Gerald's Column by Gerald Fitton

So what shall we talk about his month? Firstly PipeDream 4.5. The philosophical content of this month article concerns a ball which bounces an infinite number of times. I illustrate the bouncing ball with a Schema 2 spreadsheet together with graphs from other spreadsheet packages. I comment on problems I had in porting the bouncing ball spreadsheet between five spreadsheet packages including PipeDream, Fireworkz, Schema 2, Eureka and the PocketBook's Abacus. All the spreadsheets, their conversions and their graphs are on the Archive monthly disc.

Erratum

This one is all my fault! In last month's article I made a mistake in the second paragraph of the section headed "Drawing the Graph in PipeDream". To produce a graph in PipeDream you must first mark a block. Unfortunately I instructed you to mark C8C58; this should have been A8A58. Sorry!

PipeDream 4.5

Last month, on page 6 of Archive it was reported that Colton Software are no longer shipping PipeDream 4. Many of you have asked me why. Part of that answer is what is to be called PipeDream 4.5. It is more than a rumour; I don't have a copy but an alpha test version definitely exists.

In July 1996 Colton Software took on a "Work Experience Student" called Niall Douglas for about ten weeks (which will be completed by the time you read this in mid September) and charged him with writing an upgrade to PipeDream. Towards the end of July 1996 I spoke to Niall and he proudly reported that the new PipeDream now uses 'dynamic areas' of memory. This means: (a) It will run faster if you have a Risc PC (b) If you buy Clares' Virtualise you'll be able to run PipeDream files which are much larger than your RAM.

Niall confirmed that the cosmetic problems with running PipeDream 4 on RISC OS 3.5 (the Risc PC) have now been fixed (in particular the annoying loss of the leading part of the row numbers). Also he has added a few minor improvements such as high resolution sprites for the icons and a neater button bar. He is talking about adding a few functions which are not present in PipeDream 4 version 4.13 (but which have been included in Fireworkz) and he is looking at other suggestions from PipeDream 4 users.

I am pleased to report that both PipeDream 4.5 and (a future version of) Fireworkz will run under StrongArm. Better than that, whilst retaining backward compatibility, the new versions will be optimised (see below) for StrongArm so, if it is your intention to run StrongArm at some time in the future (and who isn't?) you should equip yourself with the new version. Some of you will wonder what is meant by "optimise". What this means specifically is that, where a compromise has to be made between the speed (and other facilities) when running on a StrongArm or non StrongArm machine, the package will make sacrifices in favour of the StrongArm environment.

I believe that StrongArm is going to transform the fortunes of the Acorn group. In

StrongArm they have a major technical development which they are marketing with real flair. I believe that any packages which are not upgraded to and optimised for StrongArm (as PipeDream and Fireworkz) will not survive.

Protechnic intend to make money out of PipeDream 4.5. So that it can be shipped with the minimum delay and in order to retain a degree of flexibility, there will be no booklets with the first shipments of PipeDream 4.5. Hence the early shipments of the upgraded version will be suitable only for current users who can get by with a disc file explaining the upgrade. The intention is to send a mail shot to all registered PipeDream users lauding the virtues of upgrading to PipeDream 4.5 and asking them to pay for the upgrade. If this small venture goes well then Protechnic's confidence in the Acorn market will increase and we will be able to look forward to further more ambitious developments.

The Acorn World Show

Colton Software will have a stand (currently numbered 116) at the Acorn World Show at Olympia in November 1996. Jill and I have been invited and we're looking forward to meeting you there. I expect PipeDream 4.5 and the version of Fireworkz optimised for StrongArm to be available at the Show and I hope that new users will be attracted to PipeDream 4.5.

Infinity

There is a popular misconception that anything which happens an infinite number of times must take forever. This may be the case when you are executing an infinitely recursive procedure on a computer but it is not true in the real world in which we all have our being!

The example which is usually used to demonstrate that this is a misconception is attributable to a man called Zeno of Elea (430 - 490 BC). As I have remarked before, I believe that much of what I teach in maths is Eternal so that if I were to use this 2400 year old example it would emphasise the timeless appeal of mathematics. Zeno's example is still as relevant and interesting today as it was when it was conceived by that 'ex pat' Greek philosopher (he lived in southern Italy). His example called "Achilles and the Tortoise" was made famous by Charles Dodson (the mathematician who wrote Alice in Wonderland under the pseudonym Lewis Caroll) and Bertrand Russell (who, in his 1903 treatise, "The Principles of Mathematics" argued that the whole of mathematics could be derived from logic). However, in spite of its timeless appeal, you must look up "Achilles and the Tortoise" (I suggest the Dodson or Russell version rather than the original) if you want to know what it is about because, to demonstrate the principle, I have made up an example of my own which I call "The bouncing ball". With complete lack of the modesty with which Paul has credited me, I would add that my students find the bouncing ball more paradoxical and hence more enlightening than Zeno's example. So Gerald's "Bouncing ball" rather than Zeno's "Achilles and the Tortoise" is what you're going to get.

The Bouncing Ball

Let me describe the scenario.

A ball is dropped from 16 feet. It accelerates under gravity. I shall assume that the acceleration due to gravity is exactly 32 fpsps (feet per second per second) rather than the true figure which is a little more. What this 32 fpsps means is that, as the ball drops, then at the end of each second it is moving 32 fps (feet per second) faster than it was at the beginning of that same second. My other assumption is that the coefficient of restitution is 6/7 = 0.857143 (approximately). This means that, after hitting the ground, the speed at which the ball leaves the ground is reduced to 6/7ths of the speed it had when it hit the ground.

The question to be answered is: "How long will the ball keep bouncing?"

The formulae I shall be using were first developed by an Italian astronomer who, somewhat unusually, is referred to by his Christian name rather than his surname. He had trouble with the Inquisition and is famous for dropping weights off the Leaning Tower of Pisa. His name is Galileo Galilei (1564 - 1642). Those of you into the physics or maths of falling bodies will be able to check all my formulae but, for those who don't want to look them up or work it out here is a summary:

The time taken for the ball to fall from a height of h when in a gravitational field having an acceleration g is =SQRT(2*h/g). In the screenshot of the Schema 2 spreadsheet below you will find this formula as =SQRT(2*B12/D@3) in the cell C12 giving the answer 1 second. The ball takes 1 second to drop the 16 feet before it hits the ground for the first time.

<u>립</u> (8)	BounceS at 100%				-
=2*SQRT(2*B13/D03)					
C13	<u>√ × = Σ f0</u>				
	A	В	C	D	
1					
2	Coefficient of restitution =			0.857143	
3	Acceleration due to gravity (in fpsps) =			32.000000	
4	Initial height (in feet) =			16.000000	
5	Total time of 100 bounces (in seconds) =			12.999997	
es.					
7	Bounce	Height	Time to	Cumulative	
8	Number	of bounce	rise and fall	time	
9		in feet	in seconds	in seconds	
10					
11	0		0.000000	0.000000	
1.2	1	16.000000	1.000000	1.0000000	
13	2	11.755102	1.714286	2.714286	
14	3	8.636401	1.469388	4.183673	Ð
ф				\$	·E

The ball hits the ground for the first time at 32 fps (feet per second). When it leaves the ground its speed has been reduced to 6/7ths of this speed so it doesn't rise to the same height (16 ft) from which it fell. The formula in cell B13 is $=B12*D@2^2$. The @ sign serves the same function as does the \$ in many spreadsheet packages. It fixes the reference as an absolute reference during replication. You will see that the height to which the ball rises is not 6/7ths but, as predicted by Galileo and explained 70 or so years later by Sir Isaac Newton, $(6/7)^2$ of the previous height. This height is about 11.755 ft.

The ball rises and falls back to earth again. It takes the same time to fall as it does to rise so, as you'll see in the screenshot, the formula in C13 is =2*SQRT(2*B13/D@3) and not half this value (as in C12). About 1.7 seconds elapses between the first and second bounce.

Finally, so far as formulae go, the formula in D13 is =D12+C13 so that the time of each bounce is added to the previous total to give the total bouncing time up to the end of the bounce number in column A.

Big cells

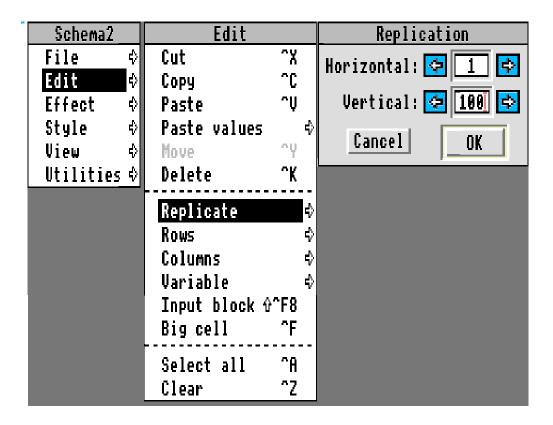
The big cell facility is Schema's way of allowing you to spread text over many columns. I have four big cells in the screenshot; these are in the block A2A5. Let me describe how to make the first of them.

Drag from A2 to C2 to mark the block of three cells and then execute <Ctrl F>. Easy isn't it? You can reverse this 'big cell' process by marking the big cell and then executing <Ctrl F> again.

The other big cells are made in the same way.

Replication

Those of you more used to PipeDream or Fireworkz may find the process of Replication of a formula in Schema 2 unusual. First mark the source cell (from which you wish to replicate the formula) and execute Copy with <Ctrl C>. This copies the formula to the clipboard. You can mark the block into which you wish to copy or you can simply place the cursor in the top left slot of the block. Then you use the pop up menus and sub menus shown in the screenshot below (Schema2 – Edit – Replicate) to bring up the Replication dialogue boxes.



The numbers shown in the "Horizontal:" and "Vertical:" dialogue boxes can be edited but, by default, they will match any block you have marked in the spreadsheet. The block into which the formula is replicated works downwards and to the right from the current position of the cursor.

If you mark a block and use Paste then you will Paste the formula into one cell only, the one containing the cursor and not into the whole of the marked block.

Returning the result

The formula in D5 is *=D111*, the cumulative time taken to bounce 100 times. You'll see that this is just under 13 seconds. Schema 2, unlike PipeDream 2, but in common with all modern spreadsheets, intelligently determines the order in which the sheet is recalculated. The fact that D5 contains a reference to a later slot in the sheet, D111, causes no problems. The formula in D5 will not work properly in PipeDream 2.

Set_value(,)

This brings me to a short digression. There is a set_value(,) function in both Schema 2 and Fireworkz (and PipeDream). However, in Schema, unlike Fireworkz, the set_value(,) function can be used only in Schema macros. It is not available for use in the main sheet. I wish it were because it is the only way of 'pushing' values from one document to another.

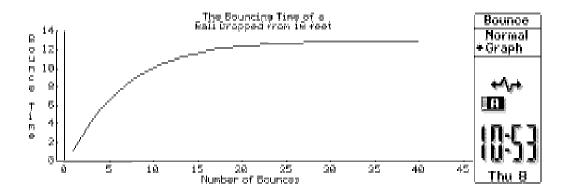
There is a set_value(,) bug in Fireworkz. Fireworkz has trouble determining the correct order in which to execute set_value(,). It works from top to bottom and from left to right. If you do not lay out your spreadsheet this way (ie revert back to 'old fashioned'

PipeDream 2 methods) then Fireworkz will show incorrect values without reporting an error. This bug has been around for some years now, it has been reported to Colton Software regularly and is not fixed in version 1.25/08. This bug is not present in PipeDream – so why is PipeDream more 'intelligent' than Fireworkz?

The set_value(,) bug of Fireworkz is not present in Schema 2 because you can't use set value(,) in a Schema 2 spreadsheet but only in a Schema 2 macro.

The first forty bounces

The screenshot below is not taken from Schema but, with apologies to Audrey Laski for encroaching on her most winsome column, it is from my PocketBook. Please do not judge the quality of the PocketBook's screen display by this screenshot. This graphic is a sprite of limited resolution whereas the two graphics from other spreadsheets (shown below) are draw files having a much higher resolution.

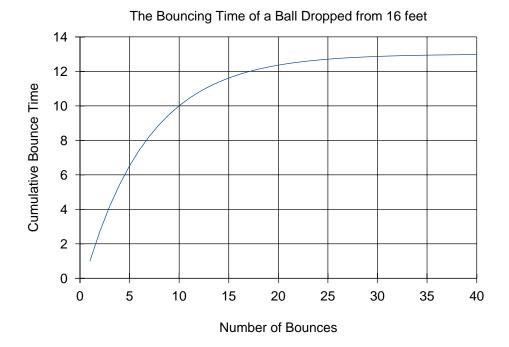


By the way, Psion are doing very well these days. It looks as though they will have a combined PocketBook mobile 'phone combination available before the end of the year. I see that Nokia are already marketing one called the Nokia 9000 for £950 with Fax and Internet software built into it. I bet the Psion version will contain an ARM chip and a lot more software. You can already link the Psion machine to any ordinary 'phone socket, get it to dial up numbers and store or display incoming data.

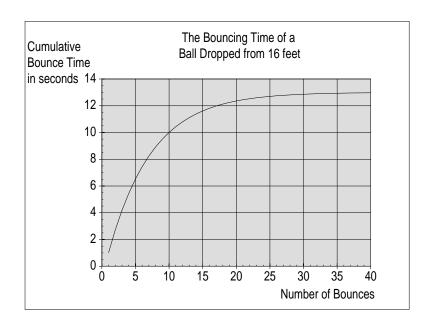
The graph above and the two below show what happens during the first 40 bounces. As you might expect the time between bounces gets smaller with each passing bounce so that whilst the first 19 bounces take 12.25 seconds the next 80 bounces take only 0.75 seconds altogether. How long will a million bounces take? How long will an infinite number of bounces take?

Ponder on that whilst I digress into Eureka, PipeDream and the portability of spreadsheets.

The screenshot below shows the same graph generated in Eureka.



Finally, my favourite form of the graph appears below:



Of course this last graph was generated in PipeDream.

Portability

I am sure that you will have gathered from the above that I have ported this [Bounce] spreadsheet, formulae and all, from one package to another. I shall return to the subject of portability in more detail another day but, for now, I want to describe the method I've used in outline only and to list, somewhat briefly, my major problems but not all of them.

I have used as my common format the WK1 format. This format is sometimes referred to as Lotus 123 format. On the Archive monthly disc I have included the [Bounce] spreadsheet in six formats. These are Lotus, PipeDream, Fireworkz, Eureka, Schema 2 and the SPR format of the PocketBook. The convention I have used for the file names is best explained with an example. The file name [BounceP] is the PipeDream version of [Bounce] and [BouncePL] is the Lotus format file which I have saved from [BounceP]. Eureka can save in Lotus WKS as well as WK1 format so I have named the WKS version as [BounceELS].

During my investigations I discovered that Fireworkz fails to convert slots which include a \$ in the formula. The \$ to which I am referring is that used in many spreadsheets (including Fireworkz, Eureka and Excel) to fix a slot reference as an absolute reference during replication. If you have Fireworkz then try this experiment. Save a document in Lotus format containing a formula which uses a \$ and then Load the Lotus format file back into Fireworkz. The formulae seem to be OK but they don't work; this is because they are formulae no longer – they are text strings. Stuart Swales (of Colton Software) is aware of this and it will be fixed in a later version.

Saving Schema 2 files in Lotus format did cause me some grief until I deleted the graphic (the chart) from within the sheet. Also you will need to open the directory containing the Schema to Lotus converter (so that the path to the utility is known to your system) but there is no need to load it to the icon bar. It is then possible to Save the file in WK1 format using the File – Export sub menu.

Eureka and the PocketBook gave no problems whatsoever. There is no need to delete the graphic (the chart) from these packages before saving in Lotus format. As a by-the-way Eureka will also Save files in Excel 3 and Excel 4 format, the only package of the five which allows this.

Fireworkz and Eureka will load WK1 format files automatically. If you use Schema 2 then you will need to open the directory containing the Lotus to Schema converter. Once you have done this Schema will load WK1 files automatically. If you use PipeDream then you must load the Lotus converter supplied with the package to the icon bar and drag the Lotus file to that icon.

The Lotus format file Saved from PipeDream will load into Eureka but will not load into Schema 2. I don't know why but, if you can track it down then please let me (and Colton Software) know. I suspect that it is something to do with the fact that Eureka will handle both WK1 and WKS formats of Lotus files.

Finally, I used to have Lotus on my PC but it seems to have 'got lost' when my son David installed Windows! As a result I am unable to load the WK1 files into their native package to see what if any are the differences.

To summarise, I give Eureka the highest marks for bug free and simple portability. Fireworkz is my number two; you can load Excel 4 files and the \$ problem will be fixed.

Determinism

I find myself flattered by the many generous remarks in your letters which refer to my

comments about the Infinite and Eternity. The letters which please me most are those from correspondents who tell me I have aroused their interest in these intriguing subjects for the first time. As I have said before, I do not wish to encroach on Paul's 'God slot' nor do I wish to offend the religious nor the atheist. However, I must report that I have already received a 'phone call from someone who is delighted to find a rationalisation "from a scientist" (me!) of his belief in free will. I don't think I have said anything revolutionary. Amongst the scientific community these days determinism is considered to be outmoded but, rather than proving free will, many see the alternative to determinism as chaos. A quote I use when introducing Statistics to my students includes the immortal lines "The race is not to the swift . . . But time and chance happeneth to them all". This quote is from an ancient writer whose name, in translation is "The teacher"; in those days teachers didn't teach Mathematics (and similar subjects) but, like all good teachers still do, he peddled wisdom.

It is my experience that, as Paul said last month (about a related but different subject) once you "have faith" (in my case about the existence of a free will) "you will find that it works". In case you are of the opinion that the matter of the existence of free will is purely academic let me add that with free will comes a personal responsibility for all the things we do to the world we live in; and with that responsibility there comes a duty to determine what is the 'right' thing to do to improve that part of Eternity on which we have our greatest influence. As one of my by-the-ways, the teacher who wrote the immortal lines above includes a warning about the random distribution of rewards: "Neither yet bread to the wise. nor riches to men of understanding". The duties arising from 'free will' must be undertaken with complete and utter altruism – if you expect rewards to be distributed equitably you will become embittered.

It is as a result of letters (and a few 'phone calls – letters preferred) seeking enlightenment about the nature of the Infinite that I have been encouraged to introduce you to the bouncing ball.

The Infinite in 13 seconds

Following William Blake (1757 – 1827) I have suggested that the nature of Infinity might be found in a single point and the essence of Eternity in a single moment. Now let me mix these concepts together by asking "How long will the ball keep bouncing?" Whilst the more practical of you will question the accuracy of the model in which a ball keeps bouncing, it is clear that, within the limitations of this mathematical model, the ball bounces an infinite number of times.

Does this mean that it bounces for all Eternity? No! We can capture the infinite number of bounces within 13 seconds of time! When I first discovered that an Infinite number of repetitions need not take an Eternity I was amazed. To me the philosophical implications are breathtaking and truly wondrous. Think about it. If we can have an Infinite number of bounces in 13 seconds then it must be possible for those 'in the know' (all knowing) to execute an Infinity of things (indeed everything?) in practically no time at all.

Let me tell you briefly about the life of a photon (a minute flicker of light). According to the Special Theory of Relativity, in particular the time dilation effect, a photon lives the whole of its life in an infinitesimal instant of (its own) time. During that 'no time at all' it can travel over enormous distances, see everything, do everything, go into hibernation, and

then live again! Truly, if you were a photon you would perceive the nature of Eternity through the instant which is your lifetime. Perhaps one day, when "the silver thread be loosed", and the swing broken at the point of its suspension, then we shall all have a perception of Eternity which, like the photon's, is timeless.

The Pursuit of Truth

I have little space left to answer the question "If the pursuit of Truth was one of the early philosophies by which you tried to live your life (see last month's article under Acknowledgements), then what do you pursue now?". My brief answer is: (a) That I remember the first time I deliberately and somewhat inexpertly perjured myself in the pursuit of Justice – the truths I had spoken had been "twisted by knaves to make a trap for fools" and I was not man enough to take that; (b) I have moved on to pursue another ideal to which I would sacrifice Justice!

Finally

(a) Please send a disc; (b) My address is that of Abacus Training; (c) Many thanks for all your letters.