

WELCOME

It's true what they say, time does fly then you're having fun. It's been a year since the first issue of GPMagazine was released and it's still going-hopefully improving with every issue. I'd like to take this chance to thank everyone who has been with the magazine from the start, and for that matter those who may have just discovered GPM- your support is what keeps the magazine being written.

Please forgive what you could see as a self-indulgent article on the making of GPM, but as I doubt I will ever get to one-hundred issues or some other large landmark I think the one year anniversary edition is a good enough place to write it. Hopefully you'll find it interesting; if not, you could always pretend that you are enjoying it. Manners cost nothing you know. Damn MTV generation with your hippity hop and Sunny Delight, where's your respect?

Anyhow, I doubt you'll be the first reader to notice that I repeat myself-I seem to have around four jokes that I slightly change for each issue. And each issue I talk about how it's the last issue. And each issue I talk about something random, then comment on it. But, each issue, you've still read it, told me what you thought of it and made me feel damn good about what I've written. So, three cheers for you all. Hip hip, hooray, and so forth.

I hope to be still here this time next year, still talking the same old bollocks and reviewing games you've never heard of, but chances are with a testosterone level high enough to grow a fuller beard.

Please enjoy the issue, and as ever, let me know what you think.

-Mark

retropunchout@hotmail.com

GPM Magazine

This month's excellent front cover comes from Shirohagen (www.holster.co.uk) , who also designed the Christmas issue's cover.

Can you name every game featured on it? Answers on the last page.

Contents

| | |
|--------------------|----|
| Cover | 1 |
| Introduction | 2 |
| Contents | 3 |
| II | 4 |
| Unirally | 5 |
| Paraball | 6 |
| Little Nemo | 6 |
| Causing A Splash | 7 |
| The MAME Interview | 11 |
| Most Wanted | 14 |
| The ZOT! Interview | 15 |
| Rise Of The Fenix | 20 |
| Nazca Dreamer | 27 |
| EndGP | 30 |



The History of GPM



The ZOT! Interview



Rise Of The Fenix

All content, apart from the front cover by Mark Rowley 2005. Front cover by Shirohagen. Some images taken from other sources. Everything to it's respective copyright holders.

A special thanks to everyone who has been interviewed for GPMagazine.

In the very first issue of GPMagazine I mentioned two things in one short article. I said that it was a dream to play Super Mario 64 on an handheld (Thank you Nintendo) and that we all want the GP64, as the second GamePark handheld was being called then. Of course, due to the amount of spelling errors in the first issue it read like a dyslexic's crossword puzzle and perhaps GamePark's English translator's were so angry at my illiterate wishes that they decided that there never would be a new GP32 machine; just to spite the community. Or maybe not-perhaps we'll never know.

Every time a fake comes along, or someone rediscovers a prototype GP32 the community becomes excited again with the talk of a new handheld. The question is though, will there ever actually be one?

Good business sense would say "no", but then again GamePark have never really appeared to have good business sense, so that can't be taken into consideration. They are a rather secretive company, and through a hybrid of this and the lack of my Korean language it's hard to say whether or not they are in a position to release a new handheld. Still, we can speculate.

The GP32's sale figures have been disputed, but before the European launch was taken into consideration there were an estimated 20,000 machines in the world. This could be almost correct or totally wrong-but this is what has been estimated by a number of sources. Then of course came the BLU, which brought with it plenty of fresh meat to the fold-as well as many NLU and FLU owners getting a piece of the backlit love. So now, if we double that figure (which is about right, what with Craigix's Retro Gamer adverts giving the machine much more publicity as well as the normal rate of sales) then we have a figure closer to, well, you can figure it out, Math's fans.

It can't cost that much to produce anymore. They have repeatedly cut corners to slash the amount of cash they need to spend on each unit (with the most obvious being the BLU+) and are now at a stage where they can be made and sold for a nice profit. Why bother making a new handheld that will never be able to compete on a commercial level with the big two, when you can stick to what it does best with the standard GP32?

On countless occasions they have confirmed that there will be a successor, and that it is in development right now. Many people are in the know about what is happening, yet obviously keep it to themselves-so something has to be going on, but it shows just how indecisive GamePark are being. My fear is that with the GP32 II, they are going to miss the mark completely. They say that open source was completely accidental on the GP32, but it will be a major part of the second one. What does this mean? Will they make it even simpler for developers to make programs and provide no boxed games at all? Or will they overshoot themselves and give it far too much power-making it complex and off putting for new developers.

The question of backwards compatibility is one that should be addressed. Does it actually hold up the potential of a handheld, tying it to it's past, or does it give access to a wealth of games and emulators from the offset? While it would be nice to believe that it will be compatible with GP32 programs, if it does happen, I believe it won't be compatible with SMC's. While very cheap now, SD cards are the way forward and GamePark will most likely want to take advantage of this. Pop turd Robbie Williams (Useless fact: my ex-teacher taught Robbie, and told him to join the army. The poem on one of his albums is about him, yet the teacher adores him. Odd.) released an album on this kind of card, and more are expected to follow. This could still allow for backwards compatibility though-it would just need programs to be copied from the SMC to the new media. Of course, commercial games would be more tricky to do, but who honestly cares about Dyhard: With Infinite Stairs anyway?

Something is bubbling over at GamePark. Let's hope they tell us what it is soon.

Unicycles. Really, what are the point in them? It's half a bike and takes twice as long to learn to ride, and even then they serve no practical purpose, apart from in a circus. Even then, it's double standards-if a clown sat in half a Morris Minor for the duration of a thirty minute show it would neither be impressive or useful. As you can see, I have problems with anything like this; I think it's since I have actually seen one guy riding it up and down the street. A bit odd, I think you'll agree.

Enough with the cynicism, on with the game. Unirally is a racer that's unlike any other, as it takes the principles from the extreme sports genre and places them into a 2D environment. You are rewarded for the tricks that you can do with a few simple button presses, and given a boost accordingly, which you then need to use to beat your opponent.

Unlike, say, Super Mario Kart or to a lesser extent F-Zero, your opponent only stands as a stick to be measured with, rather than a active competitor. There is no real way to slow down the other unicycle and it will always get a very similar time, meaning that it's the tracks that you need to learn in order to master the game. You'll need to learn to change direction, build momentum to make it up half pipe walls and most importantly, land properly in order to succeed. Obstacles also exist to slow you down, and it's memorising where they are on the 20 tracks is the real challenge.

The game has a great concept; there's no doubt about it. It's fun, challenging and rather original, but however hard I try to like it there's something holding me back from giving it a large score. It's hard to



Game Name: Unirally

System: Super Nintendo

Emulator: OpenSNES

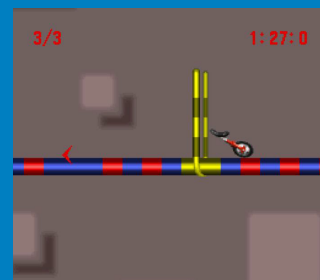
Other Notes: Full speed with very little overclocking, frameskip 0. Controls difficult to map.

One Two Three Four Five Six Seven Eight Nine Perfect Ten

define, but the fact that it feels, oddly, like a Mega Drive game more than a SNES game goes against it. I have nothing against the Genesis obviously, but it does lack the polish that I've come to expect from the Super Nintendo. The graphics are large and brash, yet bland, and it doesn't control as well as it should do.

The half-pipe sections that just require tricks to succeed aren't as much of a diversion as they should have been, as the normal game essentially revolves on this principle, but with a point A and B. The most criminal thing about this game is that it is completely devoid of all character; how hard would it have been to stick a penguin, chimp or panda on the unicycles to give it a bit of character? Super Monkey Ball managed to create some emotional attachment by sticking a primate in a sphere, and it makes the game player feel so much better.

This could have been excellent, but as it stands it is merely above average. If you want a game that really could have been, try Unirally.



REVIEW

Game Name: Paraball

System: GP32

Emulator: N/A

Other Notes: PC Port, Coded in Fenix, portable to Dreamcast.

One Two Three Four Five Six Seven **Eight** Nine Perfect Ten

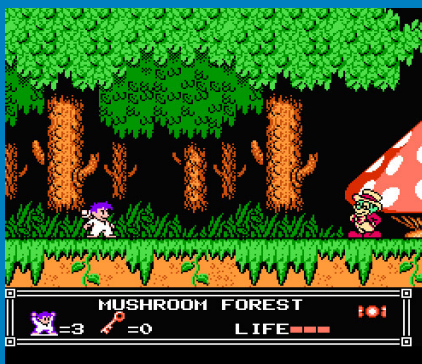


Paraball is a strange hybrid of Pong and Volleyball. I know that Pong is a recreation of Tennis, and the similarities between the two sports are obvious, but the influences from the 70's game are clear; the rinky-dink sound effects, the simple graphics and the use of the walls to score points all hark back to Pong.

The aim of the game is to simply get 10 points before your opponent does, and you do this by dropping the ball into their half of the playing field. As the player your top is rounded, meaning that the excellent physics engine understands just where the ball should go and at what speed, meaning that matches are a matter of skill instead of luck. You just use the arrow keys to move left or right, 'Up' to jump (which can prove a risky tactic, as you need to time it perfectly to get the ball to go where you want) and, in what I can only describe as an impressive innovation, you can use bullet time.

Bullet Time can be turned on and off at the touch of a button, and there is a counter of 100 that counts down when activated to stop you using it permanently. When back in Real Time, it starts to recharge. While the graphical effects of slow motion are great, it serves a real purpose, as it allows you to save shots you would of missed if it were not used.

It would be great with some sort of high score, or league mode. But as it stands it's still a huge challenge in hard mode, and well worth downloading.



Game Name: Little Nemo: The Dream Master

System: NES

Emulator: LittleJohn 0.4

Other Notes: Full speed with sound with 0 frameskip at 133mhz. Slight decrease with VSYNC.

One Two Three Four Five Six Seven **Eight** Nine Perfect Ten

If you like your games surreal, then this ones for you. At first glance it's a nice looking basic 2D platformer, but nothing more. Then, as soon as you reach the first animal and realise that when you feed it three times you take over it's body, it soon becomes apparent that this is one fantastic game.

Gorilla's mean that you can smash everything, lizards let you walk up walls, and moles let you dig. You'll need to use all of these skills to find the keys hidden through the level that open the door at the end. It's annoyingly hard in some places, and the platform engine sometimes fails you, but the magic is certainly there.



Causing a Splash? The History of GPM

Can one year really be classed as a history? I think it depends-did enough happen in that year to write a decent sized article about, or was there nothing of interest to note? I think with GPM, there is a history that has developed, so as I've said, please forgive the self-indulgent article.

Pie?

When I received my GP32 in early January 2004 and was starting to get to know the people who inhabited the forums, I thought what better way to contribute than make a website? (Well, there's a few, but none that I'm really capable of) so GPM was born-or rather, the GP32 Pie was born. Possibly the worst name for a website ever somehow seemed like a fantastic idea to me, although I grew tired of making it as soon as I realised the design was terrible, and the whole purpose of it (to be a resource for commercial games) would be an expensive venture. This is where the idea of a magazine became more appealing.

When you make something that is more 'physical' than a website it feels very good to see your work-GPM gives me that feeling when each issue is completed. The move to PDF was a logical one too; a website would be hit with heavy traffic in the first few days, then not be visited again, plus, I'm hardly a great coder. The name 'The Unofficial GamePark Magazine' was coined, which most people shortened to either GP-Magazine, GPMagazine or plain GPM-all names which I started to use down the line. I started a preview edition to give people an idea of what would be in the magazine entitled Issue Zero, but soon realised that it wouldn't be long before I could have a full issue out, so the idea was scraped.

Darts

At this point I emailed Guy Fawkes. It read as follows:

Hello there.

I reviewed your darts game on GP32x, and as you can probably tell, I'm a huge fan of it. I'm starting work on my GP32 Magazine/Fanzine (currently called GPM) and was wondering if an interview on your future projects etc. would be okay? If it is, please email me and I'll send you a few questions.

Thanks very much

-Mark

Much to my surprise I received a prompt reply where he offered to not only conduct a interview but to also send me a WIP version to preview. This was hugely encouraging- the guy whose game I loved was offering to help me out and give me the story I needed. You know that Dartboard on the front cover of the first issue? He even sent that, to which I applied a ripple effect to-that's the extent of my artistic talent.

When reviewing games for this issue I noticed I was going to run out of content quickly. There were only around twenty commercial games at this point, and while there was home brew some of it can't be reviewed as there's so little to say about it (a high percentage are great games, but are hard to fill a page of writing with) so I came up with an idea. What did everyone buy their GP32's for? Simple; they brought it to emulate the classics. And along with the classics everyone knew about, what about the obscure games no-one ever heard of?

The design of the first issue was truly horrible, and by my standards that's saying something. The yellow on gradient blue just didn't work at all-both visually and technically. My PDF writer wouldn't print the backgrounds; hence the first release as a collection of JPEG's in a ZIP file. A printer (both ink and PDF) version was then made a few days later, which someone commented looked better than the horrible colour scheme of the first release. I felt that they were right, and the gray on white design of the first three issues was due to this one comment.

Feedback

I never expected feedback as positive as I received-people were being very nice, and giving me constructive criticism which I hoped I eventually addressed. I started work on the second issue pretty soon after, and that's where possibly the biggest interview thus far took place-Blue Angelo.

Most GP32 owners will of at least heard of Blue Angelo at this point, and it could have been seen to be the savior of the GP32's commercial side. With lovely hand drawn graphics and something that appealed to many GP32 owners; RPG style leveling up, we had one game to look forward to. I sent the developers of the game an email, and received one back. To my surprise, they were willing to do it.

Or rather, they weren't. It was a case of mistaken identity- Shibuya Games were the publisher, not the developers. The interview



still revealed some details on the game though, and was one something I didn't expect; fanzines don't get to interview commercial companies.

Other than that, the second issue also had an experiment I conducted where I asked a few gamers on a forum to see how many of them had heard of our beloved handheld. It also saw the first edition of 'Most Wanted', which has been in the magazine ever since.

At this point I put together a 'Double Issue' that was never released; simply the first and second issues put together with a few mistakes corrected. What the point in doing this was I can't remember, although I know that it cut out the more obscure parts of the magazine (mainly the E-Book and MP3 Reviews).

Chat

A board member known as Narmak had offered me some hosting and a domain, so gpm.techme.org was born, and never updated. He then went AWOL for a while, although this didn't matter as much to me as Matt Magoo had kindly purchased me a .org.uk domain (meant to be a .co.uk one, although he hit the wrong extension) and I purchased some hosting and created the basic website that I've never changed.

Issue three had the announcement of the results of the competition I held that had few entries. Still, the entrants were pretty good and unique, and to them went a couple of small prizes. Super Mario War, which was a feature in the third issue, is sadly yet to appear.

The first couple of issues could be described as bland and sloppy at best; the headings hung randomly on the pages, the colour scheme was boring and the reviews were short and badly written. I had decided to take more care in the making of this issue, writing longer and more focused reviews, better articles and making the design slightly cleaner. Nigel Brown also contacted me and offered to send me a chatboard to review, which of course I accepted and based the issue upon. This was something the community actually wanted to know about and that I could deliver, a rarity indeed.

People wanted more colour in their magazine, and that's what I gave them in the fourth issue. I introduced some blue.

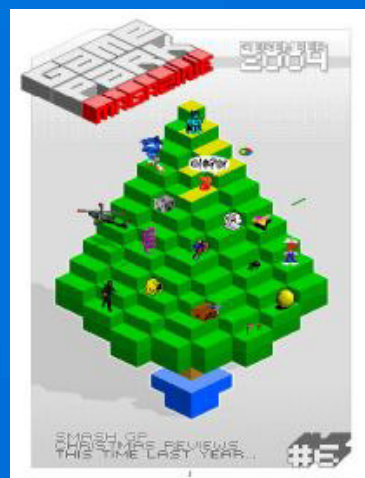
Blue

Yes, blue. The grey borders were made a light blue, and some bold text was made a darker shade of

blue. I somehow thought that this was the most colourful design I could accomplish, and that's how it stood. I've since lost the original files that made up this and the next issue, and since PDF2TXT just won't work with my magazine's they won't be rewritten any time soon.

It focused on the GP32's Mod Scene (with a front cover design I subconsciously stole from Edge) and had an interview with Sam Fisher, in the days when he was a true spammer. Spray and Dozer also wrote an article each (the only time someone else has written for GPM) which leads me nicely to Pintsized.

Matt Magoo asked me if we should make a version of GPM that can be read on the go. I'd had the idea before, and he himself had attempted to convert it, but we didn't get very far. This though, was to be a reviews-based diskmag written by Matt, me, Dozer and Spray. It got good feedback, and then seemed to disappear.



There were talks of it coming back, but I doubt it will.

The fourth issue had been completed, and someone described it as the best issue so far. I really do disagree; I'd say it's one of the weakest (although the review of Tee-Putt was a highlight of the magazine-it's one of the finest games I've ever played, even if I can't remember it's name) but if the readers like it, stick with it.

Colour

Yes, more colour. The fifth issue (with an excellent, yet random front cover a friend drew) was short at only 21 pages, but was so bright it could make your eyes bleed. The lack of reviews from the last issue was made up for with all the NES games rated, and a YoYoFR interview just after the release of OpenSnes 0.3 was I'm sure enjoyed by most, including me. The interviews really are what keeps the magazine going, and as I say I do very much enjoy them.

The review style was odd, and almost a return to the first issue's style-but not quite. I was happy with it anyway, and released it around a month after I had finished writing it. I also posted about a 'Best of' edition, and even started it, but again was never released.

Dual

At this point, the DS was becoming a reality, and I wanted to make a GPM style magazine about the DualScreen. I brought a domain, and DualScene was started (far before the dualscene.com, I believe) to be released on the GP32x forums a few days after starting writing it.

The problem was a lack of an audience, as the potential users of the NDS were so spread out across the Internet. It made me appreciate the large and accessible user base the GP32 has, and therefore GPM has. That's what, along with my love of Christmas, made the sixth issue the best so far in my opinion.

I won't talk it much about it as it was so recent and is probably still fresh in your minds, but it had a better design, seasonal theme and a feature on Smash GP. Also worth noting is the simply wonderful cover Shirohagen designed.

Next

Everytime I finish an issue of GPMagazine, I'm happy to see people enjoying it. I've said it so many times now, but there wouldn't be a point in me writing this unless at least one person had a good time reading it. The classic line is "I make games/write articles/do whatever for myself, and if anyone else likes it, all the better" and while I do make the magazine that I would like to read if I were someone else, I don't do it for myself. That would be pointless.

Expect many more promises about a 'Best of' edition and a Diskmag soon.

It's only been one year, and in a years time I'll look back on this and cringe in the same way I look at Issue One now-but...

Thank you for reading, folks.

SCORE<1> HI-SCORE SCORE<2>
0480 0000



It's here. Some said it would never happen, people were flamed, but now it's here. And, I'm sure you'll agree, it's been worth it.

Since Ryleh released the first version of his emulator for GBAX last year, MAME has grown in leaps and bounds. It can now run more than three games-many, many more than three games, and best of all, can run them well.

The following interview is with Franxis, the coder behind what is one of the most impressive things to ever come out of the Korean handheld.

Hello there, can you tell us a little about yourself?

I'm a 30 years old guy living near Madrid (Spain). I was 3 years updating an spanish emulation web page. I was bored with the webpage and i started to find some other hobbies and interests, and though I would be interested on program something for emulation scene. I buyed GP32 console last summer because it was distributed in Spain.

After some tests with GP32 programming (i ported a Columns clone i programmed for the university for MS-DOS years ago), I did a Multipac emulator port for GP32 last Addic2004 compo... The next thing would be MAME...

What's currently on your SMC?

I have 4 SMCs. 2 SMCs are dedicated to homebrew games and PC ports (Doom, Quake, Rise of the Triad, Monkey Island...). The third SMC is used for emulators and roms for each of the systems (NES, Genesis, MSX,...). The fourth SMC is for my programs (MAME, Multipac,...).

Mame GP32. A simple question but I'm sure one that many of us are thinking-how on Earth did you manage it?

I expend December 2004 studying, cleaning and simplyfing source code. On january 2005 i did the port of MS-DOS specific libraries, and at the end of january i had the first results (Ghosts'n Goblins and Pacman, yuhuuuu!) xD.

After this point is relatively easy to add compatibility to any other game.

What made you want to port Mame? Do you have any favorite arcade games?

I have expected a MAME port for months. I buy my GP32 to play old arcade games, and also console emulation.

I saw Rlyeh was porting MAME for years and didn't see to release to the public... I did a port of MultiPac (the pre-mame emulator by Nicola Salmoria) for Addic2004 Compo and i ended the project with less problems i expected (a month before the time limit, i expent 3 weeks), then i started to investigate MAME with a lot of morale at the beginning of december 2004...

My arcade classics? Pacman, Ghosts'n Goblins, Bubble Bobble, Commando, Tokio, Xain'd Sleena, Snow Bros, Black Tiger, Psycho Nicks Oscar, Bomb Jack, Tiger Road, Toki, Pang, Tetris... I had a lot of games i wanted to play on GP32 ;-).

Mame's code is notoriously bloated-where did you start, and wasn't it off putting?

Yes! A lot. The source code of MAME 0.34 are 15 Mb of C source code. I expent a lot of days cleaning code... Now MAME GP32 has 'only' 9 Mb of source code.

What would you say was the main problem you had during the process?

The main problem is really the GP32 console. It is a hell to do a decent sound player, and also to get MAME running without vertical scanlines and sound-clicks. Also I expent a lot of time optimizing code, and doing cheats to get MAME running at decent speed.

The different FXE's has led to a few people asking about some sort of front end-are there any plans for this?

The use of different fxe files is absolutely necessary, due to the limited GP32 memory size (8 Mb). I had no time to do a main frontend of all games for 03/05/2005 - MadriSX / MadriDC party. Maybe in the next version, but i think it is not foreground now.

This might sound like a stupid question (I'm not a Mame expert by any stretch of the imagination) but why the 0.34 romset?

I decided to port MAME 0.34 due to:

- It is an old version i remember to run quite well on PCs with similar power to GP32.
- It was the first MAME version with wave output FM, it didn't need a sound blaster fm chip for Ghosts'n Goblins music in example.

Then the MAME 0.34 romsets are the consequence of using this MAME version.

Rlyeh is also porting Mame to our beloved handheld; have you spoke to him about it, and do the two ports share any code either of you have written?

Yes, i have spent a lot of time speaking with Ryleh, and also he has helped me with some problems. But i don't know what are his plans with his mame port. If he decides to colaborete actively to improve my port he will be welcome.

How do you feel about public support of the emulator? The boards haven't been this lively in a very long time.

I'm very happy with it. Maybe my MAME port will make GP32 more popular. I have seen news about my MAME port in all emulation webpages today (I have not seen it before) ;-).

What was the atmosphere like at the meet up when you unveiled it?

I showed some photos a month ago and i show my first results to some friends. They were shocked (as me when i saw Ghosts'n Goblins running slowly). The show to the public was during last adriSX/MadriDC party in Madrid (Spain) 5 march 2005, a day before the Internet release. People who test the emulator during the party were surprised with the speed and compatibility of the emulator.

Most people assumed the GP32 would never see a version of Mame. Most people assumed we would never get a full speed Genesis emulator. Some people assumed we would never see a GBA emulator capable of running commercial games. The GP32 is really being pushed beyond all our expectations thanks to some great developers, but how far do you think it can go?

Maybe a perfect SNES emulator... Maybe i will try to help with this...Maybe the limit will be a perfect MAME? Or maybe a NeoGeo CD emulator but i think it will be not very usable without music...

If you could have one Mame game running on the GP32 that doesn't now, what would it be?

Tiger Road.

What would you say will be the furthest that MAME on the GP32 can go? If the majority had their GP32's RAM upgraded to 32MB, would you consider utilising it?

If i had access to asm z80 (Reesy) or 68000 (Cyclone) cores maybe the main project will be the GP32 memory ram size...I don't have such ram upgraded GP32... If someone want to sent me one i will use it immediatelly, and i think a lot of more games would be compatible without problems.

Do you have anything planned for the next release? Do you think 2 player RF link up will ever be implemented?

I don't want to program RF 2 player option, because nobody seems to have RF. I have planned to improve speed and sound quality. Also i will try to add more compatible games, but it will require an FXE file per game for sure...

Thanks very much for your time, do you have anything else to add?

Thanks to you, Mark. I will try to improve the MAME port, nothing more ;-).

Most Wanted

It's all in the title.

"Rubbish Baps"

An old man recently came into the shop where I work. When asked if he was buying baps, he said yes, but the last ones he got from there had been 'rubbish', as they had raisins in them.

Play as the pensioner as he mistakes teacakes for 'rubbish baps', and postboxes for 'rubbish toilets'.

It's a bit like: Being ever so confused. Bless.

The Good: Thinking about it, teacakes are really just rubbish baps. You wouldn't slide a burger between two, would you?

The Bad: The European cousin of the bap; the roll, also has it's sweet variety. The iced roll, in fairness, would need to be renamed if this phrase would ever take off.

Comic Relief GP

Play as a student who feels the best way to raise money for chairity is to sit in a bath full of means, cross-dress, run across a muddy field, shave their head or something else that's 'wacky'. But how much did they give out of their own pockets? Erm...

It's a bit like: It's me just being bitter.

The Fun-Bus Game

Can you resist the temptation of riding the bus? I know I can't. What will you get today? The piss soaked old lady who is always there, the single mum wearing her umbro jumper and a kid for each year of her life since she was 14, the two girls who say "Oh my God!" thirty four times in a 7 mile trip, or my personal favorite, the racist old man who tells the Asians there that they shouldn't be here.

It's a bit like: A microcosm of Stoke.

The Good: Charities are very worthy causes, and the more publicity they get the better.

The Bad: Students (myself included) are all scum. Every last one of us, and that means you.

The Good: Most people ride the bus, so I'd say that most of us have our own defence mechanisms that will help in the game.

The Bad: My defence, which involves talking backwards and fondling my own head, may not always be effective, nor suitable.

Most Wanted GP

Sit around for a while, looking at a blank screen. Then, when you finally think of something that pisses you off, instead of putting it in a diary or blog, try and think of anything remotely witty about it.

It's a bit like: Issue 2, 3, 4...

The Good: People seem to like it, and it's not that hard to write.

The Bad: They are becoming more and more obscure, and less about games. Expect me to me giving details of my dog's failing sex life in these pages soon.

INTERVIEW

ZOT!

Skeezix's Zot! Engine was always going to be an ambitious project. It takes skill to learn to talk to a computer (and it's even harder for it to listen to you) but to write the language that the computer must be force fed and make it simple enough for any mere mortal to write is something of an achievement. This challenge, however, is what motivates him.

For all of you that aren't aware, Zot! Was released into the community at around the same time that the Fenix runtime was ported. Zot can make games arguably easier than Fenix, but can still turn out some very impressive results. Then, when you are good enough at the basics, you can dive into Zot!Script to give you even more control of your creation.

The huge four page interview is with Skeezix, and when you're done be sure to check out Zot! And it's creations.



Hello there, and thank you for doing the interview. Could you tell us a little about yourself please?

Sure, start with the hard question first ;) I'm an indie coder from Toronto (Ontario, Canada) who misses games that are fast and fun - instead of overly complex games that dilute their core goal. (That's the snooty way of excusing one's self from being a retro nerd, since really we're all just happy that we can buy cool stuff from our childhood that we couldn't afford back then.) (It's also a way of saying I suck at Metal Gear Solid, but I can kick serious ass at Omega Race and Bubble Bobble.)

What's currently on your SMC?

My SMCs keep dieing (!), so only the bare essentials stay on there; I'm also one of those poor souls who tends to use a lot of his own software (so I know who to go after for bugfixes ;) since I tend to be looking up issues people have and fixing them.. really, to me, half the game is in the engine itself, rather than the playing. Anyway, on my SMC would be GP Engine (one of the older, and one of the better, emus on the GP32 :), as I'm hopelessly addicted to the music in Bomberman 94, and I'm also a fan of the TG16 (I have a Turbo Express, but it guzzles D-class batteries like you could not imagine!) and its multitude of classy shooters. CaSTaway is always eternally there naturally (though due to my work hours I don't play the big games, but I will always have a special place in my heart for Time Bandits on the Atari ST.. simple, but a true class act.) Like everyone else I've got DrMD on there, since it's just stunning and I only "relatively recently" got into the Genesis, so I'm still finding all the great originals and ports there. (And I'm a shooter fan, and Genesis has Raiden so I'm good to go. Okay, TG16 does as well.) I've also got a number of the GBAX compo homebrews, and Beats of Rage and such on there, but mostly since my SMC killed itself (for the umpteenth time :) I've been mucking with DrMD, and I really admire the Arkanoid knockoff and the Nacza stuff :)

Hmm, a year ago I whipped up a page and tried to remember some of the games I've loved over the years.. ahah, here we are. It is very incomplete and a work in progress, but it might prove amusing to anyone who reads this..

Prelude: "So anyway, this is not an exhaustive list of games I played (no one really needs to hear about Breach or Leisure Suit Larry any more..), nor is it the best games nor the worst or more influential.. it's a list of games that I find notable, that I remember, that I loved at some point or another and that I think are important." <http://www.codejedi.com/textfiles/topgames.html>

Before moving onto the Zot! engine, you have coded plenty of very well recieved programs for the GP32. CaSTaway is an excellent emulator and most people will be aware of it, however, with it emulating a computer rather than a console there will be a few members of the community who have never experienced it. How would you describe it?

An interesting question (since many GP32 folks are from the UK I'm used to things being similar to here, but we never had the Speccy or Amstrad here...) -- here of course (like everywhere), the Commodore 64 and then the PC ruled, but in between there was the Atari ST and the Amiga .. both were fairly popular here, and oh the wars we aged (your floppy clicks, your screen shakes, you Amiga bastards!)... oh, right, your question, anyway.. ;) (My after-school job for high-school was working as an Atari ST salesman, which means I have some cool Atari heirlooms kicking around :)

The Atari ST was a landmark machine.. one of the first 16/32 bit machines around but made of mostly common hardware (unlike the Amiga which was full of custom goodies.) so that it was both inexpensive and powerful; the ST had MIDI ports built in making it a favourite among musicians and gamers: the ST was the first to get crazy networking game play going. It was one of the first desktop publishing platforms (challenging the young Mac for the market), and .. had glorious games, including many that literally created the genres we enjoy today. A pretty fast machine with a lot of colours (nothing compared to what the Amiga could render, I know, shut it!) people were fairly amazed with its excellent arcade ports (Bubble Bobble, Super Sprint, etc.), early on 3d games (Star Glider, Interphase, etc.) and stunning RPGs (such as genre-spawning Dungeon Master, Captive, FATE, etc.) .. not to mention the spawning of fun BBS systems and popularizing the mouse ;) Don't get me started through..

CaSTaway is a lightweight Atari ST emulator .. designed to be portable and fast, rather than highly

accurate. As such it enjoys ports for Windows, Unix/Linux, Mac, and of course GP32 and Palm OS courtesy me. It runs most Atari ST games pretty well, though not all, and will handle most floppy based application software as well. (A hard drive is not emulated.) If you want some old school word processors, or if you *really* want to run Word Perfect spell checker on your GP32.. you*can* :)

For some screenshots from CaSTaway (Palm OS version, forgive me..) I have a page set up..

<http://www.codejedi.com/shadowplan/castscreens.html>

On the GP32, I really enjoyed Super Sprint.. always a great ST port. Of course Time Bandits plays well.. lets see.. some of the many greats:

Super Sprint, Time Bandits, Dungeon Master, Captive, Arkanoid, Xenon 1 and Xenon 2, Nebulous, Buggy Boy, Populous, Spherical, Super Hang-On, Civilization (Yes, Civ on GP32!) Sim City

Will we see anymore updates to CaSTaway?

Sure; there was a few updates just a month ago in fact, and there are some more planned when I get a moment; I have been absorbed with Zot of late, but really I've just been trying to keep up with Real Life..but I'm definately spread too thin. (I realized the other day that in one day I'd produced binaries for all of these platforms: Windows, Mac, Linux, FreeBSD, GP32, Palm OS/Zodiac, Pocket PC and Symbian. Geesh!)

I'd say CaSTaway/GP is a very complete emulator package (it even has a nice little manual ;), though I'd like to improve the savestate system to allow you to have many per game/disk, and not just a global list of 8 or 10.. too limiting if you play a lot of games, and hard to remember which is which. But overall I'd say its a very good package, with a pile of features, custom scripting, and options.

Zot! is clearly a powerful and easy to use language, could you describe it for us?

Zot was born out of my idea to make a Time Bandits-like game (TB is like Gauntlet sort of, with some twists; it started out on the CoCo.), as well as a Gauntlet-like game, and a Bomberman-like game... Realizing that really they're all variations and extensions of a theme (the tile based engine), I figured it'd be trivial to knock together a little Bomberman engine, but why not make it more generalized to see what else you could come up with? Then I realized Pacman should be doable, and then Joust, and then even Warcraft.. and so I had to see. (To me they're all a base map area, plus optional scrolling, plus things that walk around and have triggers set up so they react to certain specific stimuli....)

You may also recall at the time, the Zodiac had just popped up and the GP32 community was being all freaked out about it, and both communities were stirring up madness... Zot was one of the ways I tried to ease things over to everyone; you'd see posts from people always whining there was no eays way to make games, and from people bitching each other out over which platform they wanted to use... so here comes Zot which lets anyone write a game with no experience or tools needed, and it will work on damned near any platform around. It didn't shut anyone up, but I tried.. ;) ie: I was able to pop out a simple game demo (BombZ) in not too much time, that ran on all platforms.. this kept people from saying "you've abandoned the GP32!" or "you've abandoned the Zodiac!"..

Further, I've always done things cross platform from way back, so I've got applications for Unix, Windows, you name it.. so used Zot as an excuse to pull together a lot of code and merge it into a more coherent base, so that I can make apps for multiple devices without having to worry about all the nuances.

What made you want to write a programing language in itself? It must be insanely difficult.

Its a challenge, for sure, but its also one of those things everyone should try (if they're a masochist..). Really, config files and such are a minimal language as well, but no one thinks of them like that - your Windows .INI files are really a very specific and highly limited language, but a language nonetheless. Taking it a step further and you get a scripting language like Zotscript. (For those interested, Zotscript is a small subset of the C language that compiles to a 6502-like assembly, then I can then interpret using a small 6502-like emulator. Whack, I know ;)

So while it took an enormous amount of time to create, I expect to use it in the future, and it really does add to the flexibility of the engine *enormously*. Zotconfig (normal Zot) is a little flexible, but you still have to live within the confines of the engine goals.. but with Zotscript, you can roll out enough rope to hang yourself but-good :)

Consider.. one of the first projects being done is a topdown minigame akin to Advance Wars.. sort of a pseudo-real-time RTS/wargame. Who'd have thought it could be possible within a little tilemap dungeon crawl engine, but there you go :)

Are the games 100% compatible between platforms? (With restrictions on RAM, resolution etc.)

They are indeed; the engine is designed in a way to try and keep things portable.. ie: Right now, gamelets can only assume 2 buttons are present and an 8-way control.. this way Palm devices, GP32s, Zodiacs and Pocket PC's can all work the same control scheme. Furthermore, if a gamelet wants to use a giant map (say 4000x300 pixels or whatever) then the engine will scroll to center the player sprite.. no problem. RAM is a hard limit though.. if you go loading a pile of artwork or audio, then you know you're tossing out the smaller platforms, but thats a conscious choice of the gamelet maker. In theory, the games originally designed for such as Gauntlet can get by on very little memory, but if you wanted to write a BoR replacement, you're goign to eat through RAM like crazy.

The GP32 has a fair amount of RAM as far as Zot is concerned.

It's certainly much easier than battling with C, but as such the types of games you can create are more limited. However, from the games seen so far it seems to do them very well. Are you a big fan of Gauntlet/Bomberman/Maze games?

I only discovered Bomberman recently (first thing I tried on my GP32 pretty much ;) and was hopelessly addicted for awhile. I've been a fan of Gauntlet from way back, and own the componentry to assemble two full arcade Gauntlet machines (I collect arcade gear).. including the semi-rare Atari sticks. (The arcade joysticks are rare in that they do not rotate in socket like every other stick does, because they wanted an Atari logo on the stick top and didn't want it turning upside down!) I do love a top down maze-and-conquer game though, from older school like Gauntlet to less old school such as Alien Syndrome through to more modern variations like Baldurs Gate: Dark Alliamce on the PS2/XBox.

Could you please explain what ZotScript is, and how it builds upon the Zot! idea?

Zot is the engine itself -- it knows how to render sprites and tiles and manage things like floor/wall buttons, doors, animation and all that goodness. Zotconfig is the basic Zot language that lets you specify maps, sprites and all those goodies at a basic level, and Zotscript is the extension language that lets you customize how everything works. (ie: In Zotconfig you can specify a sprite to walk around randomly, as in BombZ, but if you want it to do certain things at certain times, you'll have to Zotscript it. ZInvaders for instance has Space Invaders that move across and down, something the Zot engine has no idea how to do on its own.)

What do you think of the response to Zot! so far-how have the other communities responded?

Until very recently, Zot has mostly been ignored and has been on the chopping block as a public tool many times (ie: I can make it less general and faster by tossing out a lot of the stuff I'd never use myself, though I just didn't want to give it up as a tool for people to use..), and I've learned that most people are just full of hot air.. though they may whine about not having easy tools to make games, when the tools show up they're "never easy enough". But thats okay, people have things to do, and it was an experiment.. but it did disappoint a little. However, more recently some folks have started to get excited.. Apex put together a website to try and help people with zot config (since it can be a little crazy at times), and both himself and a couple others are working on some little gamelets; a couple people in the mailing list have started working on a visual map and config editor, and a few others have been asking questions. So momentum is building, and theres more than 100 people in the mailing list... I'm fairly certain that once the visual editor comes to fruition that people will start fiddling more .. people are just too initimidated by editting text files by hand.

How far do you expect the Zot! idea to go?

Hard to say; its oriented towards mobile games for now, but if it keeps growing and I flesh it out more, its entirely possible it could be used to make little desktop games, or become an easy way for people to learn the basics of programming in a fun way without dealing with pointers and other scary concepts a heavy duty language will throw at you. (Or people will just go to DarkBasic and BlitzBasic :) Certainly its good for prototyping.. you can slap together a Pacman-like game in a few hours, so its a good way to see a concept and know if its halfway fun or not.

Have you tried Fenix, and if so, what did you think of it?

I've not tried Fenix; most of the info about it at the time seemed to be in non-English so I couldn't follow it, nor did it fit my needs (being easily embedded into an engine, without itself being the engine.) Fenix does seem to perform very well considering its fully interpreted, and seems to be growing a lot of applets on the GP32. Sweet stuff, and Chui gets thumbs up all over the place.

When development on the DS/PSP really comes into it's own, do you have any plans to support them?

Possibly, but not sure at this point; I've got a lot on the go, so not sure I can adopt too many more platforms.. but it very much depends on the policies surrounding the devices, and if I have any of them. ie: DS homebrew has started happening but is still at the serious hacker level (ie:you have to open up the case of the device, and Nintendo is not endorsing the activity at all), so its not something I can think about. The PSP so far is an entirely closed platform -- little is known and Sony isn't talking. If either device becomes an open platform, I'll check them out. (I've rather had it with Sony's evil practices.. they've made my life tough in terms of development many times..) -- I don't have a lot of interest in platforms that squeeze out people from maximizing their value.

(I must admit to using the PS2 Independance Day hack to boot homebrew code on my un-modded PS2 though.. running a Genesis emu on your PS2 is very fine ;)

Do you have any other projects planned for the GP32?

I have often been tempted to knock out a simple Vic 20 emulator, since so few exist on any platform ;) Theres a lot of projects, but less time :)

Finally...what does Zot! mean? ;)

The word "Zot!" is used in comic books a lot, usually when someone is being hit by something!

Thanks for doing the interview-do you have anything else to add?

I'm totally tired out right now, and look forward to making fresh muffins soon... but for me, theres a lot on the go to wrap up, many projects on the go, and wondering where things will go.. should be an interesting year in terms of finding out what the possibilities are.

Has anyone considered...

Palm OS, Pocket PC and GP32 might be the last open mobile platforms?

ie: If DS and PSP are never endorsed, then it becomes wholly up to the hacker community ("hacker" in the truest sense, the honourable sense) to turn them into fun platforms. If that never happens...

So GP32 may live forever as the last of the wild west platforms :)

Thank you for your time and the great interview. Links can be found towards the back of the magazine.

Rise of the Fenix

A few months back I, as I'm sure many GP32 owners have done, decided that I wanted to learn to code. I went to my nearest bookstore, and after being distracted by the other sections for a good half hour I found my way to the computer language area, picked up a copy of 'C++ For Dummies' and paid for it. I then took it home, loaded the IDE up on my PC and opened it at page one-just as soon as I'd had a lie down to take the strain of my back. It's a damn big book, and after a short while it crippled me both mentally and physically.

While coding was never my strong point, I knew my way around so to speak. I had my first experience of a language on (don't laugh) a V-Tech PowerPoint 'laptop' which contained a very stripped down version of Basic. I used the book that came with it to learn myself the language, and after spending a few years (literally-I didn't have a PC at this point) coding it I was doing some very challenging stuff-my favorite being the movement of a letter around a few squares at the players input. It doesn't sound impressive, but when the only way to do so was a series of IF statements, PRINT statements and a hell of a lot of CLS's, it takes a lot of code. Alas, you could only save one program at a time, which to be honest was a bit of a bitch.

Right, I went off at a bit of a tangent there, but my point is that I wasn't a newbie to the principles of coding. I'd also spent many years in The Games Factory, so the building blocks were there again. Perhaps it was naïve of me to think that this past experience would help, but I didn't think it would hurt my progress. When writing a simple "Hello World" program took about an hour when soaking everything in, my thirst for the immediacy I was used to was something I just needed.

The book has long since been put down, and lost somewhere in the drawers to my bed. The only two things I can really remember from it are the guy on the front cover who looks a bit like Violet Berlin, and that the writer was prematurely balding. I started to take notice of Fenix properly early this year, but only really started to learn the language towards the end of February. However, a few days after picking up the basics, I had a simple, maths based program working called 'Petals Around The Rose', which you can find on the next page.

It has the simplicity of the easiest scripting language, yet speed and functionality that you wouldn't expect. If you've ever used DIV you'll feel at home, and if you haven't, don't worry. Over the next few pages, I hope to give you a bit of an insight into Fenix.

Please note that I am indeed a beginner, so the code isn't as clean as it should be, and many things either won't be explained correctly or at all. Hardly a good thing, but check it out-perhaps it'll make a coder out of you.

Before You Start

Before you start to learn Fenix, there are a few things that you will need, and need to know how to use. Some are English, some are Spanish-but all are simple.

What You Need

Fenix Beta 0.84b

http://www.flamingbird.com/public_html/filemgmt/index.php

These are the files you need to run Fenix games on your PC. Make sure that you are getting this version and not the later versions, and they don't seem to work on the GP32. When you load up most Fenix software (such as FPGEdit, which can be found later in this list) they may ask for where these files are kept. It makes sense to store these files in a file on your desktop.

FlameBird 2.11

http://www.flamingbird.com/public_html/filemgmt/index.php

You can code Fenix games by just typing the code into notepad and running it through the runtime files in the Command Prompt, but this is a long and needless process. FireBird is a IDE, which means it give you an enviroment to code, and then allows you to run your program with a button click. It also checks your code and changes it's colour to help organise it, and can suggest fixes to basic errors.

FPGEdit

<http://cdiv.sourceforge.net/html/down/downfpgedit.htm>

Images used in Fenix games are stored in FPG files, which again you can make manually or use a program to make. FPGEdit is that such program.

FNTEdit

<http://cdiv.sourceforge.net/html/down/downfntedit.htm>

Just as with the FPG files, you have to prepare fonts for use with Fenix. FNTEdit allows you to take a standard font and make a .fnt file from it. These programs normally install to your start menu.

Fenix GP32 Runtime

<http://www.gp32x.de>

Follow the links through the development section until you get to 'GP32 Files and Fenix', which helpfully provides the GP32 runtime (what you need to run the games on your GP32-if you've played any Fenix games before, you'll already have this in GPSYS).

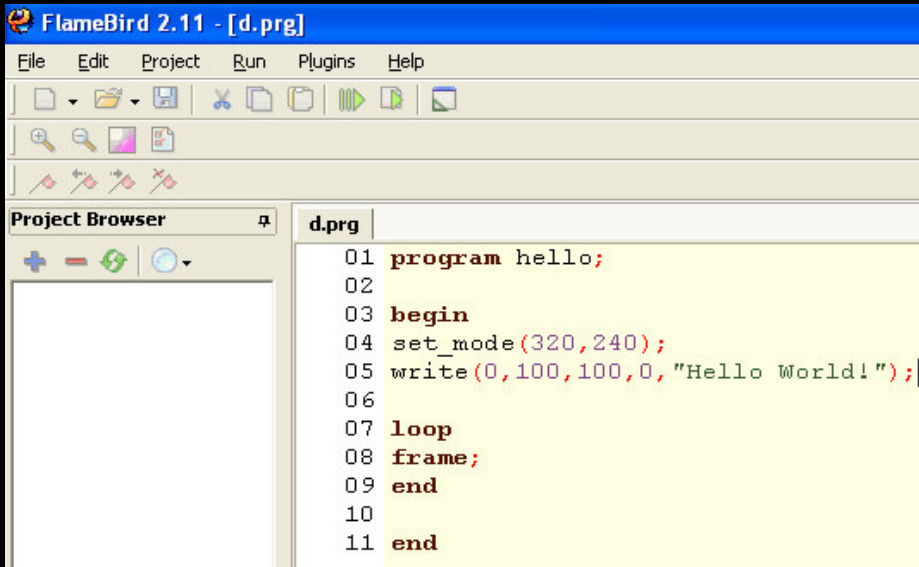
Recommended

Photoshop

<http://www.adobe.com/products/photoshop/main.html>

Even if you do't really know what you're doing with it (like me) it's still worth checking the demo out, as it will allow you to set images with the amount of pixels you need. Also, use it's blur filters to hide your terrible drawing skills.

How To Use FlameBird



The main section of white is where you type the code—it has line numbers in order to help you. It's a pretty standard affair—but if you look closely you can see how it colour codes things, for example, the words in dark green are the words contained within speech marks, and are therefore shown on the screen.

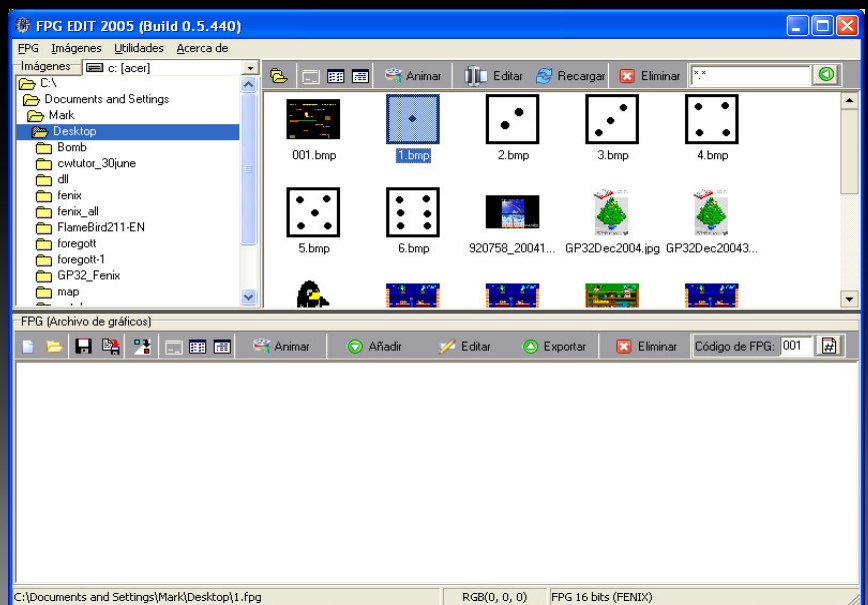
The Run command at the top of the screen allows you to compile and run (in other words, test) your program straight away. Note that some parts of Flamebird are in Spanish.

How To Use FPG Edit

First of all, press File, then new. This is completely in Spanish yet it's clear what each command is. Here you need to type the directory and the name of the FPG pack you want to make, and set the mode to FPG 16bit (Fenix). Press Accept, and your FPG is ready to be filled.

Using the directories on the left, navigate to where the pictures you want to use are stored. They will then appear in the top section, and just need to be dragged down to the bottom section and saved.

To keep the file size down, press file, and convert to the first option. Then press Utilities, and Convert To Fenix.



How To Use FNT Edit

This is so, so, so simple. Simply choose the font and size, and press generate. Select where the font is going and you're done.

For a nice, clean, simple font, I use Verdana in size 10 (which is what this is written in).

Hello World, It's Me Again

Nearly everyone's first program in a language is a 'Hello World' one, as it teaches both the most basic commands and the structure of the coding. The writing in blue is the actual code used. The **Extra** sections can be skipped until you become more familiar with the code.

```
program hello;  
  
begin  
set_mode(320,240);  
write(0,100,100,0,"Hello World!");  
  
loop  
frame;  
end  
  
end
```

Each program starts off with you saying what the program is called, just to tell Fenix what this is that you are writing. This one is called 'hello', and is then followed by a semi-colon. This is how nearly every line is ended in Fenix.

Extra: Fenix, like C, doesn't see lines in the same way that we do. Our eyes would see the word to the far right on this line as being the end of the line, but Fenix sees the ' ; ' as being the end. This means you can write multiple Fenix lines on one line that we would see.

The 'begin' command tells Fenix that you are beginning something, which is in this case the program itself.

The next line is something that we only need to tell Fenix to do once, which in this case is prepare it for the GP32. You see, Fenix games can be played on the GP32, Dreamcast, Windows and I'm sure a few others, but in order to play it on the GP32, it needs the GP32's resolution. The screen is 320 pixels across and 240 pixels up, which is where the 320,240 part comes from.

This is how most commands are set out. They are often (but not always) a word followed by an underscore, then another word. The brackets then say what you want to do, in this case, set the resolution to 320 by 240. Be sure to note the semi-colon after the closing bracket.

Extra: A good feature of Flamebird is that it predicts what you are typing, then gives you a choice of which command to select. The stuff in the brackets is also explained by Flamebird (a small tab will appear when you type the first bracket, and tell you what it needs, such as where to put certain values. If you don't know a particular value, just put a 0-it often works.)

The next line is what puts the writing on screen. The standard 'write' command is for writing text to a certain position on the screen. Then, the contents of the brackets give the details. The first number is the font you want to use (when you have no preference, the default font will do, so just put a '0' as I have done.) Then, a ',' is used to separate the values. The two 100's are where to put the text-in this case, 100 pixels from the top of the screen, and 100 pixels from the left. The next '0' is the type, which to be honest isn't important right now.

The final section in the brackets is what you want to put on screen. This must be placed within "" marks, else Fenix will think that it is pointing to a value called what you have written. Note the closing bracket and semi-colon.

If we ended the code there by deleting the rest of the code and putting an 'end' to finish it, a black box would appear on the screen and quickly close. That's because we haven't told it to wait, and show us what's on the screen.

That's where the loop and frame commands come in. When you tell the program to loop something, it will do so indefinitely. The frame command just says 'stick this on the screen', and that is what it is looping. The next 'end' tells Fenix to end the loop.

The final end is there to end the program itself. Remember when we told it to begin? This where we close that. **That's it-why not try changing some of the numbers, text etc. and see what happens? And remember-most lines finish with a semi-colon.**

Petals Around The Rose

This was my first real program in Fenix, and as such it doesn't have the cleanest code-but it still can be learnt from. It has nothing complex, yet uses graphics, if statements and processes, which you will need to use. You can find the program in the Fenix section of GP32x.com.

First of all, I made a file on my desktop called Petals, then made some pictures. I made each of a simple dice, and then a background picture which was a rose. I then made a font in FNT Edit, which I called petals.fnt, and was just size 10 Verdana.

To make my FPG I navigated the directories on the left of FPG Edit and chose the petals file on my desktop. I then dragged each file in one by one, starting with the 'number one' on the dice, and ending with the background picture. You need to make sure you drag the pictures in the correct order, 1,2,3,4,5,6, Background else the program won't work correctly. Then follow the instructions on a previous page to compress it.

| | |
|--------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>program petals;</code> | As with the Hello World tutorial, you need to tell Fenix that you are starting the program. This one is called 'petals', and is then followed by a semi colon. The next line is not followed by a semi-colon though, as it is telling Fenix that you are about to declare the global variables. |
| <code>global</code> | |
| <code>int dice_1;</code> | If something is global, it exists throughout the whole program. We will always need the dice, so I declare them here with the 'int dice_1;' command. The reason the word 'int' is before the 'dice_1;' is because it is a standard number. |
| <code>int dice_2;</code> | |
| <code>int dice_3;</code> | |
| <code>int dice_4;</code> | |
| <code>int dice_5;</code> | |
| <code>int font;</code> | Extra: If you wanted to, say, ask someone what their name was, you would need to declare a 'string' instead. For you to do this, you would need to declare 'string name_1;' or something like that. |
| <code>int randoms;</code> | |
| <code>int petals;</code> | |
| <code>int your_answer;</code> | |
| <code>int ban_1;</code> | The other's that I have declared may or may not be used in the rest of the program; some have been left from when I used them, but have now been deleted. Not the most tight programming, then. |
| <code>begin</code> | |
| <code>roll_dice();</code> | |
| <code>end</code> | I then tell Fenix that I am beginning to code the main code, which doesn't have a semi-colon after it. What we have next is a piece of code that calls a Process. It's essentially saying, "Now you're here, go and look at Roll_Dice". It is followed by a pair of closed brackets, and finally an end command that ends the main section that |
| <code>process roll_dice()</code> | |
| <code>begin</code> | |
| <code>DELETE_TEXT("Correct! There are</code> | <code>Petals.");</code> you began two lines ago. Don't worry, it's not as confusing |
| <code>DELETE_TEXT("Wrong! There are</code> | <code>Petals.");</code> as it sounds-just remember that when you begin |
| <code>your_answer=0;</code> | something, you need to end it-else it would carry on forever. Since we just told Fenix to go and roll the dice, we need to tell it how to. |
| <code>set_mode(320,240);</code> | |
| <code>load_FPG ("petals.fpg");</code> | |
| <code>font=load_fnt ("petals.fnt");</code> | The 'Process roll_dice()' line (that again, does not have a semi-colon after it) is creating a sort of box that you will be putting the information needed to roll the dice in. When the program is told to, it looks in the box. The brackets aren't important right now. |
| <code>randoms=rand(1,6);</code> | |
| <code>ban_1=1;</code> | |
| <code>petals=0;</code> | |
| <code>put_screen(0,7);</code> | You need to use a begin statement as normal, to tell Fenix that you are starting to write the process. |
| <code>write(0,35,50,0,"How Many Petals Are Around The Rose?");</code> | |
| <code>write(0,30,80,0,"Use Arrow Keys To Adjust Answer:");</code> | |
| <code>write(0,28,100,0,"Start Checks The Answer, 'A' Rolls the dice.");</code> | |
| <code>write_int(0,280,80,0,&your_answer);</code> | The DELETE_TEXT function does just that-it deletes the text that you put in quotations marks from the screen. I'm sure you've noticed though, that there is no text saying this on screen-but there will be later. |


```

dice_1=rand(1,6);
PUT(0,dice_1,55,170);
dice_2=rand(1,6);
PUT(0,dice_2,105,170);
dice_3=rand(1,6);
PUT(0,dice_3,155,170);
dice_4=rand(1,6);
PUT(0,dice_4,205,170);
dice_5=rand(1,6);
PUT(0,dice_5,255,170);
if (dice_1==3)petals=petals+2; end
if (dice_1==5)petals=petals+4; end
if (dice_2==3)petals=petals+2; end
if (dice_2==5)petals=petals+4; end
if (dice_3==3)petals=petals+2; end
if (dice_3==5)petals=petals+4; end
if (dice_4==3)petals=petals+2; end
if (dice_4==5)petals=petals+4; end
if (dice_5==3)petals=petals+2; end
if (dice_5==5)petals=petals+4; end

```

```

loop
if(key(_up))
your_answer=your_answer+1;
while(key(_up))frame;end
end
if(key(_down))
your_answer=your_answer-1;
while(key(_down))frame;end
end
if(key(_enter))check();end
if (ban_1==2) roll_dice();end
if(key(_control))ban_1=2;end

```

```

frame;
end

```

```

frame;
end

```

```

process check()
begin
if (your_answer==petals)write (0,70,130,0,"Correct! There are    Petals.");
write_int(0,186,130,0,&petals); ban_1=12;
end
if (your_answer!=petals)write (0,70,130,0,"Wrong! There are    Petals.");
write_int(0,186,130,0,&petals);
end

```

```

loop
frame;
end

```

```

end

```

The next five lines are things that set the program up, and make it ready to use on the GP32. By using the `your_answer=0` command, we set the answer the player gives to 0. This is the number it will be by default, but when we restart it we want it to reset to this number.

The next line we've seen before, and sets the window size to 320 by 240.

The next line however is new, but not hard to understand. It simply loads the FPG file with the pictures in. Make sure you put quotation marks around the file's name. The next line does the same with the font, BUT we need to make it so we can call the font when we want to. That's why we write "font=" before hand, so that when we say 'font' it loads that font.

The random variable has no point, and should be ignored. I then set `ban_1`(the name means nothing-it's just a switch) to 1 and the amount of petals to 0.

The `Put_screen` command puts a picture from the loaded FPG on the screen, and is mostly used for backgrounds. In this case the (0,7) means it loads the 7th picture from the default FPG, which is number 0.

The write commands were used in the last tutorial, except for the `write_int` command, which is the same except it shows `your_answer` on the screen. Note the `&` symbol before hand.

The next section is repeated six times for each dice. By setting each dice to `rand(1,6)`, it will be a number between 1 and 6. This is where the dice are shown on the screen, by using the `PUT` command. It puts a picture on the screen from the default FPG, which in this case is 0. It then chooses the picture's number-which in this case is equal to `dice_1`. For example, if `Dice_1` is equal to 5, it shows the 5th picture. The last two numbers are it's position. This is repeated for every dice-each separated by 50 pixels.

If you've ever completed the Petals Around The Rose test, you'll understand that for each dice with a dot in the middle, you add the dots around the edges together across the five dice to find the answer-those are the petals around the rose.

Since we can't tell this to Fenix directly, we need to use a maths formulae.

To find the number of petals you take 1 away from each odd number and add them together. Since $1-1=0$, we don't need to say that, so we just use three and five instead. This is an IF statement, which says IF whatever you type in brackets is true, then do what is out of the brackets. You don't need a semi-colon

outside of the brackets. In this case, it sets the number of petals to petals +2 or +4. In other words, it adds a few to itself.

After every IF statement, you need to close it with an End.

Next up we have a Loop that contains the controls. They are looped because they need to be used throughout the program. The first IF statment bascially says that when the UP key is pressed, add one to the player answer. When you 'talk' about controls in Fenix, it is often like this- key(_whateverkey), with whateverkey obvously being the key you want.

The GP32's 'A' button is not the same as the Keyboard's 'A' button though-below is a list of the correct keys as taken from the GP32x forums.

| | |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| key(_left) = Stick to Left | The While function isn't needed, but allows the number to move more slowly. If it wasn't included, each time you pressed 'up' the number would increase a bit too quickly. (Where as when you restart the program, or roll the dice again, it goes up in threes for some reason.) |
| key(_right) = Stick to Right | |
| key(_up) = Stick to Up | |
| key(_down) = Stick to Down | |
| key(_control) = A Button | |
| key(_alt) = B Button | It is then put on the screen with frame; , and the IF statement finished with an End. This is repeated for the down key, except the answer is decreased by one. |
| key(_enter) = Start Button | |
| key(_space) = Select Button | |
| key(_tab) = L Button | |
| key(_backspace) = R Button | The next two if statments are much simplier-they have no while statement. |

The first one says that if the enter key is pressed (or on the GP32, the start key) then go and do the check() process. Simple enough. The next one says that if ban_1=12 (don't worry, this has been randomly named and valued at 12 for no real reason) then go and do the roll_dice process-in other words, start it again. How is ban_1 set to 12? That's the next IF statement, which says that if control is pressed then set it to 12. You could of jsut had one IF statement, which could of been `if(key(_control))roll_dice();end` and why I haven't done it like that I'm not quite sure.

Important: Remember, that Fenix see's = as being something into something else. I mean, if you say boats=mangos, then it will set boats to mangos. If you say boats==mango,

It is then framed, and ended, and framed, and ended again. These are things that were opened earlier-the loop and the process, and need to be closed. Both frames are most likely not needed.

The next process is started here, which is called 'check'. It is called when you press the start button, and contains two IF statements.

If petals is equal to the players answers, then it shows the text saying that it is correct, then the number of petals. There is a large space before the word 'petals' which the answer is displayed.

However, if the answer is not equal (we use a != for this) then say that is is wrong, but still show the answer.

We then close these statements, and then the process, and finally we loop, frame, end and close the program with another end.

That's it, the program is finished.

There are a few bugs, such as each time you restart the game it moves the answer up in threes, then nines, then twenty-sevens etc. I think this is because of the positioning of the loops, but I can't figure it out. If you can, I'd be interested to hear from you. My email address is in the introduction.

Hope this inspires you to check Fenix out.

INTERVIEW



nazcadreamer

If you've just read my article on Fenix, I'm sure you will have noticed that it's a language with a lot of potential, but as I'm a beginner I can't really get the most out of it. Nazca, however, is different. He has written some truly wonderful games for the GP32, all in Fenix and with some fantastic Mode-7 effects.

Nazca Dreams may be his most popular work, but games such as his 3D Snake clone NazcaTron, and Super Mario Kart style racer Nazca Runners are equally as good and criminally ignored. If you haven't played these games yet, I recommend doing so immediately.

Nazca is making the homebrew that is keeping the GP32 scene fresh. Read what he had to say over the next few pages.

Hello there, could you please tell us a little about yourself?

I'm a spanish 21 years old, third course audiovisuals student, and a multidisciplinary pseudoartist ;) . I do paintings, comics, but I am more into music right now being the singer and songwriter of a band called Ca\$hflow, here in Malaga (yes, Costa del Sol, Andalucia :D)

What's currently on your SMC? Do you have any favorite programs/ROMs?

Let me see... uhm, TONS of homebrew games. I love homebrew, it was the reason I bought a GP32, GPspout, Lacuna, Falldown, etc. My favourite one is Oankali's Puzzle Mix, my girlfriend and me spent many hours playing it... (you know, girls and puzzles :)) but I also love emulation as well, I have every rlyeh emu and drMD. My fav roms are "Wonder Boy III" and "Terranigma". In fact I love most Action-RPGs from those years and platformers, but also I enjoy classics like "Head over Heels" and "Below the Root" so much.

First of all, can I ask what the word 'Nazca' means?

Nazca is an area, a town and a culture of Peru (you, english-speakers people call it Nasca). I always have been really fascinated with South American cultures, specially with this one. Some drawings in the ground of Nazca desert, the "Nazca lines" are famous in the whole world because of their unknown meanings. You can read more about it @ wikipedia (<http://en.wikipedia.org/wiki/Nazca>).

The little cute white guy seems to have become your mascot; where did the idea for him come from?

It's just a sketch. I draw a lot when I am boring, they are little sketches and many of them means nothing. Sometimes I just find that some artwork can fit in a game, a painting or whatever. I think Nazcadreamer is cute enough and meaningless for being my mascot, isn't it?

How did you get involved with Fenix? Were you a DIV user?

I knew about Fenix and DIV years ago but I really hadn't any interest until chui ported the Fenix runtime to GP32. The fact that you could easily code for a handheld and create a portable game is cool enough to try and learn Fenix.

How long have you been using it now?

I think I started to code in Fenix in June or July, 2004. I don't remember exactly.

What appeals to you about the language? Do you code in any other languages?

I think it's one of the best things we have in GP32 community, it's so simple and powerful. We can play a MOD or draw a background just with a line of code.

Anyway, sometimes a C hacker come to me and tells me: "you're not a real devver,man. Developing must be f*****g hard!". LOL. Yeah, I'm not a devver. Not everybody wants to become a profesional developer, some people -like me- only want to learn something new and see what can they do with some free tools. Fenix is perfect for us. It has a powerful yet simple syntax, suitable for the novice and more or less usable for the experienced.

I know some C basics, not much, but I can actually printf on screen "halo weird" ;D.

Your games on the GP32 are all pretty unusual and original-how did this style develop?

Really? Great! :-) I don't think I have my own style, just I do the games I want to see on GP32. I always used the same tools for every game, same palette, etc. Actually, I didn't mind about palettes, I just used Windows Paint's standard. I work in 8 bit colors, drawing pixel by pixel with the 8x lens tool. Simple, huh? I have moved definitely to Linux and I use Gimp instead Paint.

What would you say is the favorite of the games you've created so far?

I think it's Nazcadreams. I put a lot of heart on it, and I think people understood it despite the inexistent argument and simple tasks. Come on, you were able to glide in dreams, it's something so catchy and lovely you can't disagree ;). I'm happy with that game, It's just what I was thinking of when I was coding it.

I think it's worth talking about the music in the Nazca series-some of it's just brilliant. The Nazca Dreams tune is especially beautiful. Where does it come from?

Some tunes are mine, some are friends' works and the rest I have no idea. The main tune of ND, I really don't know who did it. It came in a old free MODs compilation cdrom with no reference to his author, It's a shame I can't give credits to him. If anybody knows who is, please let me know. :/

Mode7 is very rarely seen on the GP32-yet all five of your big games include it, and most (such as Nazcatron) rely on it. Are you personally a big fan of the Snes-Style trick?

Yes! I am a big fan of Snes hardware and software. I think it was a superb machine ahead of his time, and I still remember those scalings, effects and 2D tricks! they were so cool!

Unfortunately, our beloved GP32 Fenix port isn't very fast when doing mode7, but if you keep an eye on speed you can do neat things. Surely it's a very limited technique, pixelated and all, but it's so easy to understand and master in Fenix.

How do you feel about the response to your games so far? How did it feel to score so highly in the GBAX compo?

Honestly, I'm very glad people likes my work. I receive a lot of feedback wich is a good thing, this means people have interest in my games. It was great the third position in GBAX compo, there were a lot of good software and I didn't expect to win a prize. Thanks again for your interest, people! :)

Are you working on anything else at the momment?

I have been working in an Action RPG some months, it's a non-linear one with huge and beautiful world, some original creatures and some hilarious dialogs. Unfortunately I have not much spare time for coding, so please, don't email me asking when I will release it.

I also was working in an audio app, a loop-machine like SonicFoundry's ACID called Nazca MusicBox. It has a bundled tuner for ocarina (my girlfriend gave me a great one for christmas) and guitar. And finally, I would like to code something for Anarchy's (www.gp32spain.com) next coding compo... but that will be if I have the needed time :/

What advice can you give to a Fenix coder who is just starting out?

Er... uhm... PATIENCE! and a spanish-english dictionary. LOL. Weird enough, seems that there is no much english tutorials and examples. I know EvilDragon is writing one, and I am doing another as well. I want to put comments to my sources and release it as (bad)examples for the community.

Start doing small things, and be open-minded, don't think there is only a way for doing something. If you are really stuck, you can drop me a line to my email ([nazca -dot- cashflow -at- gmail -dot- com](mailto:nazca-dot-cashflow-at-gmail-dot-com))...and try to enjoy coding!

Thanks for your time. Is there anything else you would like to add?

Thanks to you for your great work, Mark. Keep the GP32 scene up people, and enjoy!



<http://www.gp-magazine.org.uk>
<http://www.gp32x.com>
<http://www.gp32x.de>
<http://ahuh.free.fr/paraball/>
http://www.talfi.net/gp32_franxis/
<http://www.codejedi.com/zot/>
<http://www.gp32spain.com/nazca/>
<http://www.flamingbird.com>

Answers To The Front Cover Search: Dig Dug, Galaxian, Donkey Kong, Centipede, Bubble Bobble, Bombjack, Pacman, Ms Pacman and Space Invaders. Raise your hand if you fell for Shirohagen's trick and only said Pacman, and not the feminist version.

I thought so....