GAMASUTRA
The Art & Business of Making Games

PRINT

Indie Postmortem: 'FishEd'

By Andy Roberts

In the Beginning

After working in the games industry for fifteen years, I'd become tired of the treadmill, the corporate environment, the clichés, the licences, and indeed, the apathy and blandness that seemed to pervade most corners of the industry at the time. Looking for new avenues and fresh pastures, in 2004 I formed a new company, Scary Fish Ltd.

In August of that year I began learning the Blitz+ programming language, the aim being to produce a PC version of a classic Commodore 64 favorite of mine. I needed to re-invent my love of games from the inside-out, and this seemed like the best way forward; I would have *complete* control over every aspect of the project, and the ability to dictate exactly when the project should be released. The indie scene is increasingly overlooked by the industry it once spawned. I guess I just wanted to redress the balance.

Author Andy Roberts

During the research phase, it rapidly became clear that I'd need a map editor of sorts to put the levels together. However, having spent days scouring the internet for likely candidates it became clear that there wasn't a single worthy contender. Ironically, the community forums were full of would-be programmers who'd also started writing their own editor, but most were only half-finished and thus fell woefully short of the mark.

It became apparent that the next generation of bedroom coders all went through the same process, that of re-inventing the wheel, creating half-baked editors instead of their magnum opus. The ratio of programmers versus the number of games being produced was (and still is) completely unbalanced, and the toolsets on offer to 2D game developers are *exceedingly* thin on the ground.

My motivation was therefore two-pronged; I could produce a comprehensive map editor so that the indie community could focus on making their games, and during the process I could learn the Blitz+ language, an invaluable primer for the skills I'd need to produce my up-and-coming game.

Plus, I'd never worked on a Windows application or an editor before. “Hey, this could be fun,” I thought, blissfully unaware of the nightmare/adventure that lay ahead. And thus I began working on "FishEd".

Research, Research, Research

The first phase of the project involved researching the competition. Plenty of editors existed for putting together Game Boy maps and tilesets, and these were often some of the best of the bunch. They were, however, completely redundant for anything other than Game Boy development.

I paid particularly close attention to Open tUME (http://members.aol.com/opentume/), particularly as I'd used the program on previous projects and it seemed exhaustive in its list of options and features, and I also rounded up every single map and tile editor I could find. The aim was not to leech ideas, more to find out what was lacking. My biggest inspirations, however, would turn out to be Photoshop and my old favourite, Deluxe Paint. The emphasis here would be on an intuitive, user-friendly experience. I wanted an artist-friendly editor, *not* a programmer-centric tool.

What emerged after over a year and a half of part-time programming was something I'm immensely proud of. Towards the end of the project, FishEd became a lesson in the importance of dedication and perseverance when tackling even the smallest of projects, and it's fair to say that completing the project, at times, became more important than the actual end result.

What Went Right:

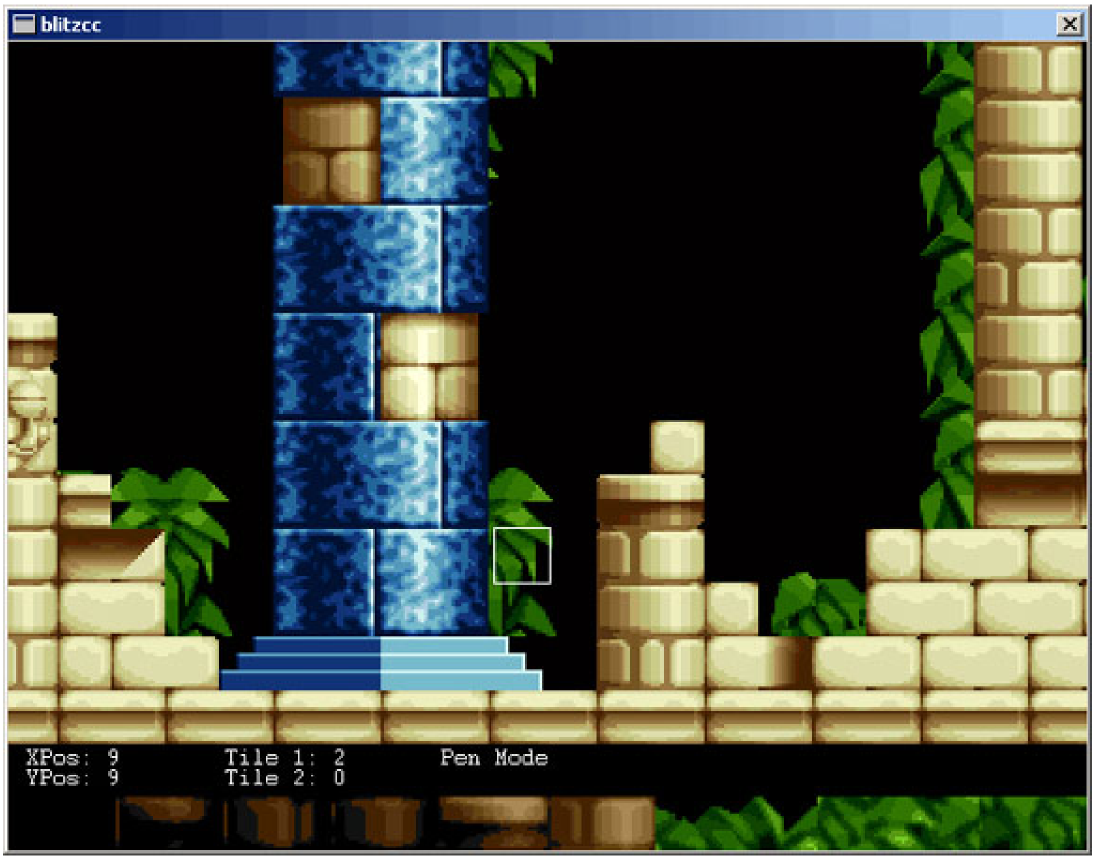
1. The Language

In my far-from-illustrious programming career, I'd previously only worked on two systems: the Commodore 64 back in 1993, and the PC in 1994. In both cases I was programming in pure Assembly language, a considerably different kettle of fish from other languages such as C (I'd never been able to get to grips with the C / C++ coding style, much preferring the logical, step-by-step nature of assembly). Learning Blitz+, therefore, was a *huge* leap out of my comfort zone; not only was I programming again after some ten years, I was learning a new language in a new environment.

I needn't have been so cautious, however; the Blitz family of languages are extremely easy to pick up (possibly more so than the likes of C), and the process of loading graphics and getting them up onscreen was simpler than I ever could have imagined. Blitz+ also benefited from the addition of a GUI module – one of my main reasons for choosing the language – and this allowed me to get the basic framework up and running after just a few hours at the keyboard. This freed up more time to concentrate on the nuts and bolts programming.

2. Research & Planning

Having worked as a Senior Designer for a number of videogame companies, I was used to spending the first week of a project trawling through websites and making copious notes. It was important to not only ensure that my planned features list contained all of the usual functions and drawing tools but to also take the user experience a step further. With this in mind, every editor I found was given a rigorous test; any quirks, problems, or annoyances were duly noted. It was particularly important to avoid the mistakes others had made.



XPos: 9
YPos: 9
Tile 1: 2
Tile 2: 0
Pen Mode

The very first version of what would become FishEd.

After a couple of weeks building the framework and fitting in the basics, it was then a very straightforward process to build up the Wish List, containing all of the features which other editors didn't do particularly well. Dozens of smaller ideas, tweaks, and refinements also made their way into the finished product, but the Wish List subsequently became the project bible, containing all of the major tasks necessary to produce what I would consider a definitive editor. This list was also useful when I hit a brick wall – tasks were categorized according to how much time they should take, so when problems surfaced I could temporarily switch to another small task to retain project momentum.

What Went Wrong:

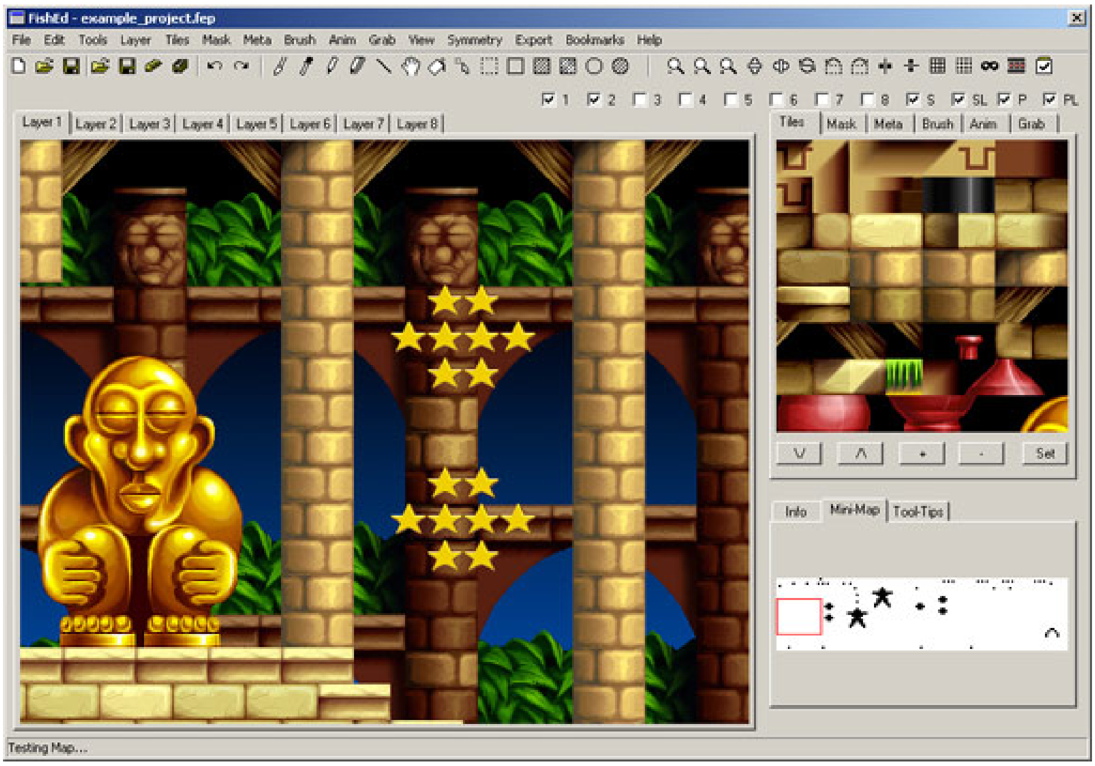
1. Coding Inexperience

Despite my industry experience working on hundreds of projects and meeting thousands of deadlines, I'd never been in the position where I'd had to program under pressure. All of my past programming experience had been within an extremely flexible freelance framework, and like many hobbyist programmers I was simply way too comfortable in old habits and working practices. Planning a schedule was one thing, keeping to it was entirely another, and at times I suffered severe whiplash watching the deadlines whooshing past.

This lack of experience was particularly evident when bugs surfaced. The support of friends and forums can only extend so far, and the more complicated routines would be impossible to post online for evaluation. Coupled with the fact that there was simply no time scheduled in to fix bugs, addressing problems and making tweaks was left until the very last minute.

2. The Wrong Language

Roughly half-way through FishEd's development, just as the project was really beginning to hit its stride, I learned that Blitz Research were releasing their latest creation, Blitz Max. While the language itself wasn't a quantum-shift from what I was used to, the Object-Oriented approach was. Furthermore, faced with an increased schedule and thousands of lines of code to convert, it seemed like I'd already gone past the point of no return and would have to finish the project in Blitz+.


FishEd - example_project.fep
File Edit Tools Layer Tiles Mask Meta Brush Anim Grab View Symmetry Export Bookmarks Help
☑1 ☑2 □3 □4 □5 □6 □7 □8 ☑S ☑SL ☑P ☑PL
Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 | Layer 6 | Layer 7 | Layer 8
Tiles | Mask | Meta | Brush | Anim | Grab
V ^ + - Set
Info | Mini-Map | Tool-Tips
Testing Map...

The final version of FishEd.

While this didn't impact the finished product too heavily in terms of planned functionality, it did mean that many features had to be hard-coded rather than relying on Blitz Max's rather gorgeous array of additional graphics capabilities. This ultimately tied up valuable time and resources, and was the source of a number of extremely stubborn bugs.

3. Lack of Testing

Despite receiving a lot of interest from various programmers and friends who were interested in testing FishEd, real life usually took priority and thus the bulk of the testing fell on my shoulders. While this wasn't particularly difficult in theory, given my past experience in QA, in practice it was often very hard to uncover bugs due to the fact that I was so close to the project and thus had written the program around my preferences.

The net result was a plethora of bugs which would only show up by accident, usually when writing documentation, modifying existing code or adding new features. Though most were easily fixed, the release date was pushed back twice due to these unforeseen issues.

4. Time Management

With any part-time endeavor, discipline, determination, and stamina are three vital ingredients, and what started off as a fun hobby project quickly turned into something much more serious. Though there was no shortage of ideas and the willingness to implement them, the simple fact remained that there are only 24 hours in a day, and many of those were often spent working on other projects to pay the bills.

In 2005 I relocated from the UK to Canada, and as a result the project schedule lost approximately 6 months over the course of development, pushing the release date back even further than originally intended (in retrospect some of the lost time would have been particularly useful for in-depth testing and addressing bugs).

5. Documentation

I'd never really understood why companies hired dedicated Technical Writers until I actually had to sit down and create the program documentation. After almost three weeks and two rewrites, the answer soon became clear: writing documentation that is functional, informative, and accessible for users of any proficiency, is an extremely specialized task, and a world apart from the manuals which accompany games.

The first draft was an extremely basic overview accompanied by various menu and GUI details, and was naturally often lacking when it came to fine detail. I took great pains to sift through the documentation for countless other applications. My end result was a hybrid of various techniques and structures which seemed to stand out and work well, but the end result took far longer than the weekend I had originally (and rather naively) penciled-in.

Conclusion

FishEd was finally released on August 1st 2006, almost two years since its initial inception. It had always been a dream of mine to create something from scratch entirely on my own, taking a paper design right through to a finished product. Despite my videogame industry experience, nothing could have prepared me for writing a fully-fledged application, and it’s painfully clear to me now why companies need dedicated Tools Programmers.

While those who helped along the way are modest about their involvement, I was very fortunate to find the right help just when I needed it, even if it entailed collaring someone via Instant Messenger at 2am to ask about obscure graphics problems. Working alone is one thing, being receptive to outside help is entirely another.

As far as the future is concerned, I'll be re-writing FishEd in Blitz Max, which will also allow Macintosh owners to use the program for their projects. I'll also be integrating Level Editors and Enemy Editors into the package, opening up the scope of the program even further. I’ll be sure to re-read this Postmortem before I start…

'FishEd'

DEVELOPER:
Andy Roberts, Scary Fish Ltd.

CONTRIBUTORS:
Paul Maskelyne,
Mark Hennessey-Barrett,
Paul Snart, Stoo Cambridge

BUDGET:
Approximately £2,000

LENGTH OF DEVELOPMENT CYCLE:
Two years

PROJECT LENGTH:
20,200 lines of code

RELEASE DATE:
August 1st, 2006

TARGET PLATFORM:
Windows

DEVELOPMENT WORKSTATIONS:
2.5Ghz PC / Win2k,
1.5Ghz Laptop / WinXP

DEVELOPMENT TOOLS:
Blitz+, PhotoShop,
PX Binary Viewer,
Nullsoft Install System,
Axialis Icon Workshop

Return to the full version of this article