Postcard from GDC Europe 2005: Postmortem: SCEE's *WipEout Pure*

By iain simons

WipEout Pure

Dave Burrows and Martin Linklater of SCEE gave a postmortem of Sony Liverpool's PlayStation Portable launch title, and the latest in the venerable *Wipeout* futuristic racing series, at last week's GDC Europe. With a relaxed presentational style, Burrows informs us that this isn't an instructional presentation, but more a general postmortem of the team's experiences on *WipEout Pure,* with the hope that there might be something useful for people to take away. And there was.

Beginning with some statistics, he gives a broad overview of the *WipEout* brand, that most 'Playstation' of Playstation titles. To date there have been six different regional versions of *WipEout Pure*, which have sold around 450,000 units - illustrating that comment with some shots he took at midnight the previous night (that being the time that PSP launched in Europe).

Waiting For Hardware

Much of the story of *WipEout Pure* is about coping with the absence of finished hardware. Preproduction began on the game in August 2003 with just two staff members, production beginning proper in October of that year. It wasn't until a year later, in August 2004 that the team actually received development kits. Finally, in November 2004 they finally received the browser which forms a core element of the download functionality of the PSP. By launch, the team had grown to twenty.

Dave, one of those two original staff members, began with a brief postmortem of the previous title, *WipEout Fusion*. He detailed the important lessons that they had learned from their experiences on that title. One of the principal difficulties he cited was that artists and designers were fighting almost all of the time: "The situation was that the designers would design a track, and then hand it to the artists, but then designers would then want to tweak the designs all the time. The tension arose mainly however, because a bad build process." Dave cited an external editor they used for the game
as a core problem - with revisions taking many hours. These time delays made the tweaking that the designers wanted and needed to do incompatible with a smooth workflow.

Added to this, within the studio they also found themselves using two tools, making ships in 3ds max and tracks in SoftImage. The result was specialists in one or the other, but few people who knew both - indeed, the one programmer who knew the external editor became a huge bottleneck. The conclusions that they came to, moving forwards, were radical. They decided on a clean start, rebuilding the game from scratch.

From Zero To Playable

One of the core needs for the *WipEout Pure* developers were that they needed to get to a point in development where they could actually play the game as soon as possible. "We needed the dynamics, user interface, A.I. and core elements first so we can get in and test. Not in a finished state, but in a state that is at least basically playable", they commented as part of the lecture. This methodology spilt through into the development iterations, which became based around six-week cycles. "If you're going to do weapons, instead of spending six weeks on just the bombs and getting them perfect, we spent three days on each weapon to get them all working at least in a fashion - and then go back over them." Dave's Powerpoint slides illustrated this approach perfectly - "Don't care about graphics! Do care about processes!"

The art & design pipeline also required attention. Again, Dave's slides cut to the chase - 'NO CUSTOM EDITOR!' The *WipEout Fusion* team found 50% of their time being soaked up in building a UI for their editor, which frequently suffered from the threat of scope-creep. As they wanted a very quick turnaround for export, enabling a much quicker testing of track designs, the decision was made to create plugins for Maya to achieve this. This custom plugin creates a track from any type of spline and is built around a very expandable exporter - exporting the entire Maya scene hierarchy.

Dave went on to demo the track editor in action. Within five minutes, an entire track had been created and was being played. The 450k file that is created that contains everything - polygon subdividing, AI data and reset polygons around the track. This is a radical leap forwards, cutting down the

WipEout Pure

entire turnaround time from five to six hours down to just five minutes.

Easy Coder?

The code design targets for *WipEout Pure* were very clear: add structure, remove specialization and create 'code empathy'. Dave remarks, "It doesn't matter who did the code, anyone should be able to know roughly what is going on." A highly object orientated structure was arrived at, with all game modules all being connected within this hierarchy. Every game module is inherited from the linkobj base class, contained within a core tree structure.

In order to play the game as soon as possible, new solutions were found for the UI development. The entire UI solution was authored in XML format, allowing for very easy revisions to be made to any content. This also abstracted the designers away from the programmers, giving them a freedom to create away from the codebase. The downside to this, of course, was that the XML file become somewhat large.

The download system was an essential part of the game, added to showcase the PSP's capabilities, provide something new to *WipEout,* extending the game content, and, of course, potentially developing extra revenue. The items that would be available for download are the obvious candidates: tracks, ships, music, skins and billboards.

Not iTunes, But *WipEout*

Dave discussed at great length the detail of how to bypass the problems involved in getting the PSP to recognize the download files, which needed to conform to the valid game save data file format. The solution eventually required the team to piggy back their game data onto a valid PSP system entry, using a dummy PSP data file. The problem of actually getting the content onto the device was related with amusing candor. "At first, we thought we might make an iTunes-type application", explained Dave, "but then we realized that it might be a little too much hassle to build iTunes as
well as a new *WipEout* game so we settled on a web-based Java applet."

This has proved to work very effectively, empowering the player to be able to add to the game either using the browser or a memory stick. The speed at which the game can be updated has been a gift for the marketing department, with customized versions of tracks being regularly made for promotions, previews and free download gifts.

Its success has clearly signaled the pathway towards revenue-raising episodic content models, perhaps releasing new content every month. While this has clear benefits in extending the lifecycle of the game, it isn't without its drawbacks, as Dave explains in his closing remarks: "Aside from the obvious resource implication, the artists hate it. Essentially for them, the game is never finished."


More *Pure*

Return to the full version of this article