

PRODUCTIONTOOL ROUNDUP

Building the perfect

Optimising and automating all those boring, mundane and time-consuming build, management and production processes will give you more time for the fun stuff, reckons **Jon Jordan**...

It would be going too far to say the technology we've brought together under the somewhat loose term 'source, build, management and process tools' is becoming sexy in the way that engines somehow effortlessly seem to be. What is certain is that really smart developers are spending more time ensuring they have a rock-solid framework in this area.

The reason is simple. The level of efficiency required if you hope to get your game project completed on time, on spec and on budget can only be met if you know what your dozens or hundreds (delete where appropriate) of developers are doing.

The problem with this, particularly from a management point of view, is akin to Heisenberg's uncertainty principle. Too often, the act of trying to observe what's going in destroys what's going on. To that degree at least, project status and project momentum cannot both be known.

And that's where lightweight tools such as Perforce and Hansoft come into play. By allowing the common-or-garden developers to get on with their tasks in an as unencumbered way as possible, the more confidence managers can have in their higher level tracking. Yes, maybe it's time to finally ditch the Post-It notes.

PERFORCE

TECHNOLOGY:

Perforce 2008.1

CLIENTS:

Activision, BioWare, EA, Epic, Midway, Monolith, Remedy, Sony, THQ, Ubisoft, amongst others

PLATFORMS:

Windows, Linux, Mac OS X, Solaris (clients)

INTEGRATION WITH

3ds Max, Maya, Office, Photoshop, XSI

COST:

From \$900 per user

CONTACT:

+44 845 345 0116



Perforce is the industry-standard for coders

www.perforce.com

The industry-standard package for source control (aka software configuration management), Perforce features a centralised repository for assets, which are accessed by client software, including users operating from

remote locations or those with irregular internet access. It's designed to operate as seamlessly as possible, and is scalable in terms of being able to handle terabytes of data and thousands of users.

SOFTIMAGE

TECHNOLOGY:

Alienbrain 8.2

CLIENTS:

Bizarre Creations, Blizzard, Codemasters, id, EA, Rockstar, Sega, Sony, THQ, Ubisoft

PLATFORMS:

Windows, Linux, Mac OS X (clients)

INTEGRATION WITH

3ds Max, Maya, Photoshop, Visual Studio, XSI

COST:

Starter pack from £12,000

CONTACT:

+44 1404 831715



Alienbrain is designed to enable artists to get in on the act

www.softimage.com

Designed as an asset management for artists, Alienbrain uses a central server to store assets and data. It handles version control and enables you to add non-destructive annotations, while tracking tools mean leads and

managers can easily review a project's progress. Integration is provided with most 3D art packages, and the latest service pack adds supports for such 64-bit applications, as well as a 64-bit SDK.

MOGWARE

TECHNOLOGY:

Mog Complete

CLIENTS:

Bioware, Glypfx Games, Sensory Sweep, Trilogy Studios, amongst others

PLATFORMS:

Windows (2000/XP, Vista)

INTEGRATION WITH

Unreal Engine 3, DAZ 3D

COST:

\$799 Complete, \$199 Library

CONTACT:

info@mogware.com



Mog Complete feature three pipeline tools

www.mogware.com

Mog Complete is a suite of individual server-based collaborative products designed to streamline data management and production workflow. Mog Pipeline provides the ability to automate data conversion processes;

Mog Library is a lightweight version control system for artists; while MogBox is a processing chain that uses a visual flowchart-style interface to enable you to handle and automate complex data conversion.

beast

HANSOFT

TECHNOLOGY:
Hansoft v5.2

CLIENTS:
Blitz, CCP, Creative Assembly, EA, Eurocom, Ninja Theory, Red Mile, Starbreeze, Ubisoft

PLATFORMS:
Windows (2000/XP, Vista)

INTEGRATION WITH
Available through SDK

COST:
€25 per user per month,
€33 server fee per month

CONTACT:
+46 18 10 90 90



Hansoft is great for tracking agile methods

www.hansoft.se

Swedish tools provider Hansoft continues to pick up high profile clients for its real-time production management tool.

The latest version of Hansoft beefs up the QA functionality, with better integration for agile development a

particular focus. The tool supports different agile methodologies such as Scrum and eXtreme Programming, enables remote working with 256-bit encryption, and allows you to set up various hierarchical planning processes.

XOREAX

TECHNOLOGY:
IncrediBuild v3.31

CLIENTS:
Over 300 including Big Huge Games, Codemasters, Crytek, DICE, id, Relic, Ubisoft

PLATFORMS:
Windows

INTEGRATION WITH
Visual Studio 6.0, .NET
and 2005 platforms

COST:
\$349 per client license

CONTACT:
sales@xoreax.com



IncrediBuild uses distributed parallel processing

www.xoreax.com

Based around distributive or grid computing, IncrediBuild is a technology you can use to speed up heavy-duty processing tasks such as Microsoft Visual Studio builds, art and AI preprocesses, QA scripts and

other data builds. The Xoreax Grid Engine creates a virtualised, distributed parallel process across any PCs in your office. Xoreax claims build times can be decreased by up to 90 per cent using IncrediBuild.

Concurrency



by David Jefferies

Black Rock Studio



WHEN HERB SUTTER WROTE 'The Free Lunch Is Over' in early 2005, his article on multi-core programming, it seemed that a revolution in programming was on its way. Single threaded programming was dead and we were all going to have to get used to mysterious race conditions and suffocating memory management.

Then, earlier this year, Jim Wilcox published his counter-article – 'How 'bout that Free Lunch?' – where he argued, from a very PC centric point-of-view, that the revolution simply never came.

He claimed most programmers continue to write single threaded code and the gains of a multi-processor environment come from having Windows and other applications running on a separate core to your own process. He also points out that programmers get 'free' gains by using concurrent middleware such as Microsoft Windows Presentation Foundation.

So how does this relate to current generation hardware? The Xbox 360, with its three cores and unified system memory, is most similar to what Herb was originally talking about – those mysterious race conditions are all around, but if your programmers are disciplined enough then it's not too worrying.

At least, it's not worrying compared to the PS3. The PS3's co-processors can only see 256KB of RAM each, compounding the concurrency issues with a big memory management headache. The data to be processed needs to be squirted onto the Cell in tiny packets and then squirted back when it's done. This is when Herb's article really starts to bite.

I don't have any hard evidence for this, but I suspect most games leave the majority of their Cell processing up to middleware such as Havok or FMOD and do their main game processing in a single threaded manner, perhaps using one of the Cell co-processors for easy-to-parallelise modules such as particles. In other words, most of us are doing what Wilcox claims.

For those of us that are trying to utilise the Cell it really all comes down to memory management. If we can organise our data into discrete chunks suitable for processing by a Cell co-processor, and we can guarantee that no other code is going to access that data, then it's plain sailing. Of course, that's the difficult bit – especially when you've got a tonne of legacy library code scattering reads and writes all over the address space.

So we've written a protected memory class that gathers all the data needed by a module, and with all reads and writes going through it's interface we can assert if the memory is being accessed at the wrong time. At this point it's important to get the migration strategy right: we're never going to convert our whole codebase at once and, indeed, some of it – such as HUD and game logic – isn't suitable for this scheme at all.

We identify suitable candidates for concurrency using the profiler and then bit by bit, class by class, module by module we move our code over to it. Looks like Herb was right after all.

www.blackrockstudio.com