One for all

The kingdoms of art and programming can – and should – happily be united under one wise digital asset management system, according to **Dave Robertson** of Perforce…

The golfer Gary Player once said: "It's funny, but the more I practice, the luckier I become". Work on the form, and the substance will follow.

Equally, many of the more consistently successful players in our industry have invested in tools and techniques that reduce the effects of luck as much as possible. Asset management (AM) now forms the foundation of these systems, but many studios still decide to implement separate art and code systems.

Can these groups be reconciled to one strategy? A united AM system needs to be functional for all the different kinds of assets and groups of users (artists, coders) and stakeholders (third party IP providers, QA specialists, publishers and hardware specialists). It needs to support indirect communication (so people can see what's going on without necessarily having to ask Jack, Jill and Mary) since such transparency underpins collaboration. Finally, it must be adaptable and trustworthy in terms of integrity and transaction control.

What is software? A collection of files on a disk. What are digital assets? A collection of files on a disk.

You see the point. If all graphics and software are digital assets, but not all digital assets are graphics or software, it makes sense to base a UAM system on technology with the ability to manage any kind of assets. Enter Software Configuration Management (SCM).

SCM IN PRACTICE

There are various ways that game developers are using SCM solutions to create UAM systems. Cambridge-based Zoonami has built a proprietary *max* plug-in for its SCM system and Lodestone Games has developed a *max* plug-in that it has put into the public domain.

Digital Extremes, in contrast, stores the 'tip' of the binary revision of the game, which enables people to synchronise quickly without having to run installers and copy stuff over the network, or distribute CDs. The studio stores the art assets of the game in their original form (in a 'contents' subfolder in the asset repository) as well as in converted game-friendly form (in a 'game' subfolder in the asset repository).

All of Digital Extreme's artists use the Windows GUI of their UAM system. When an artist has finished working on an animation, say in *max*, they use a propriety export tool from within *max* that converts the file into a game-friendly format and exports it into the game's content folder, so they can submit it easily into the asset repository.

The artist can then sync this version of the file they have been working on into the latest version of the game on their desktop. In this way they can almost immediately see the result of the work they have done, be it anything from animations and textures to sound. If the results are good, they can submit the file/s to the asset repository by simply dragging and dropping.

When doing so, they submit both the converted version into the game branch and the original source art into the 'content' branch. This is done because the game conversion tends to strip out all non-essential aspects of the artwork to be as compact as possible when running the game.

OVERNIGHT ACTION

Recently, the process has been further automated to reduce the number of export steps – shortening the artist pipeline is a big payoff. A daily overnight build of the game handles the small percentage of changes that can't be immediately synced into the game; if significant changes are occurring, a build will take place once a day as well.

When submitting files, level designers and artists attach a brief description such as: "The crazy monster running into the wall on level three has been fixed." These notes are then posted to an internal group to keep people up to date. An alternative would be to automatically send emails detailing changes being made.

In conclusion, a game development project is much like any other R&D effort. By taking care in choosing your UAM route, you needn't risk alienating any section of your creative talent.

■ **www.perforce.com**

PHASED BUT UNFAZED

How difficult is it to change your habits?

"Irrational Games Australia originally chose *Perforce* because it provided a robust version control system for the programming team. However we decided it could be the best solution to all our asset management problems.

To smoothly transfer to a unified system, we adopted a three-phase plan:

Phase One: *Perforce* was installed for the programming team to test for reliability and support during the prototype phase of the project. Our initial investment was for six clients and the server

Phase Two: A limited test was carried out by the design team. This required that the programming team modified some of our tools to handle version control through *Perforce*. The integration of the system into our development tools was relatively straightforward. At this stage we purchased two additional clients

Phase Three: The entire team switched over to using *Perforce* to manage levels, documents, sounds and art assets for the game, and we purchased 14 additional clients.

By month nine of the project, every asset for the game, including levels, documents, art assets, sounds, even concept art, were all stored in the *Perforce* repository and accessed via *Perforce*."

Tony Oakden, producer, Irrational Games