CMP
United Business Media

PRODUCT REVIEW AUTODESK 3DS MAX 8 SOFTWARE

FEBRUARY 2006

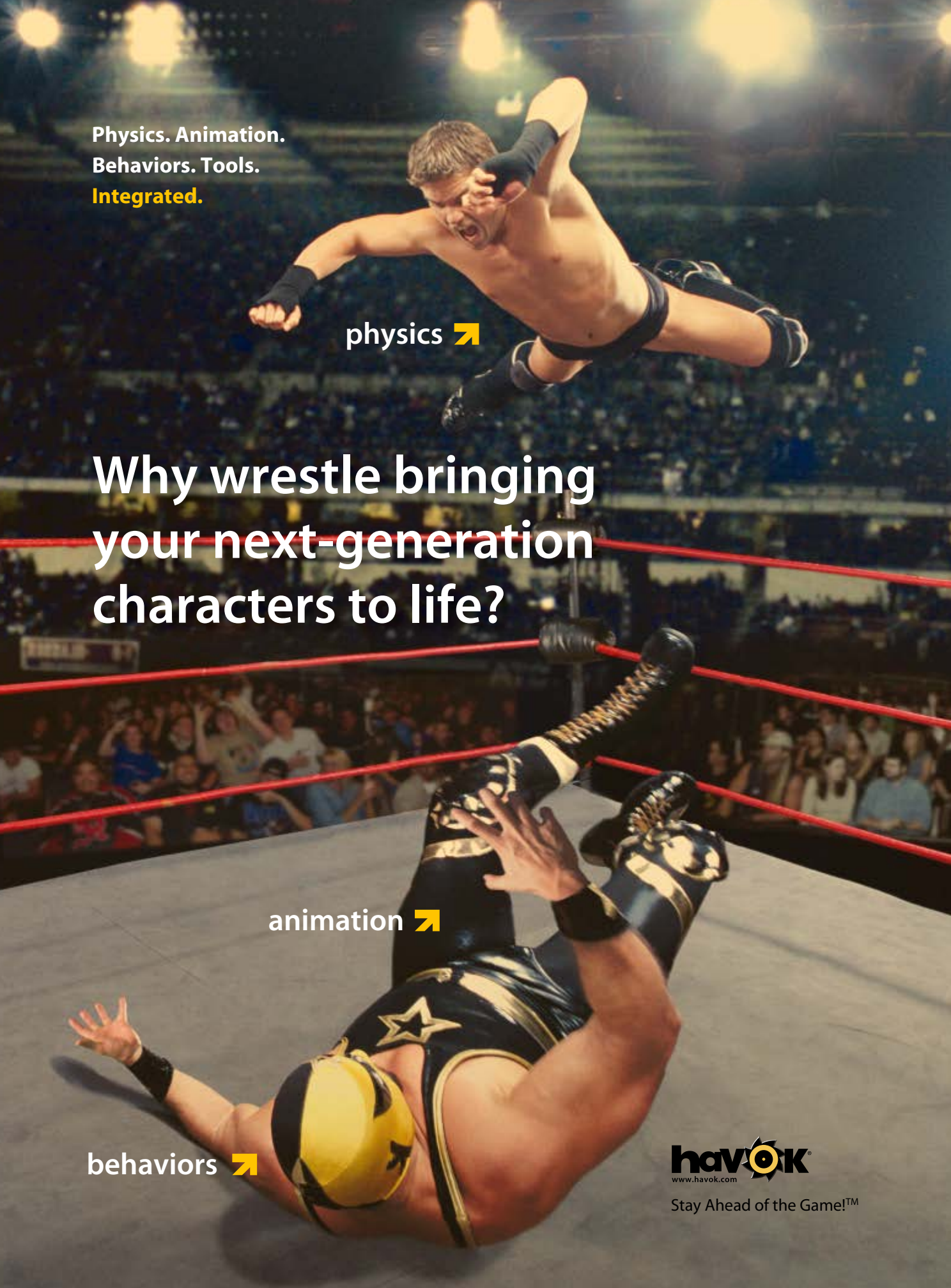
gamedeveloper

THE LEADING GAME INDUSTRY MAGAZINE

STATE OF THE INDUSTRY
CASUAL GAMES AND THE MASS MARKET

SIT! SPEAK! DEVELOP!
MAKING JAK X WITH NAUGHTY DOG DISCIPLINE

DEMOLICIOUS DERBY
A CORNUCOPIA OF TRIAL TOOLS FOR ARTISTS


Physics. Animation.
Behaviors. Tools.
Integrated.
physics
Why wrestle bringing your next-generation characters to life?
animation
behaviors
havok
www.havok.com
Stay Ahead of the Game!™

gamedeveloper

FEBRUARY 2006
VOLUME 13, NUMBER 2

CONTENTS

FEATURES

POSTMORTEM

DEPARTMENTS

COVER ART: RYAN LESSER, HARMONIX

COLUMNS

HERO TO MOST

THIS MONTH'S GAME DEVELOPER IS, WELL, NOT afraid to rock out a little, as you can see by the jaunty cover, which features a postmortem of Harmonix's PlayStation 2 mainstream rock phenomenon, GUITAR HERO. GUITAR HERO has been consistently eulogized since its release late last year, and received a significant number of votes in sister site Gamasutra's best games of 2005 survey.

To play devil's advocate for a moment, why does GUITAR HERO particularly matter? It's just another of those darned rhythm games—isn't it?—except with that guitar controller and a few more bands that we've heard of.

No, no, and no. One thing that games forget to be is culturally appropriate, and, at least for the Western market, GUITAR HERO has caught the zeitgeist dead on.

If you add classic rock poster artwork, actually skillful cover versions to riff against, plus careful, not overly intricate note-picking—which really makes you feel like you're Jimi (or Eric, or Lemmy)—then you're in the money. It's not about how clinically games are created so much as the way they make you feel, and the folks at Harmonix explain how they make us feel real good (see pg. 24).

CASUAL WHISTLING

A general theme for this issue is the widening of the game market, something that's readily apparent in GUITAR HERO. Feature writer Paul Hyman, however, takes a look at one of the areas that's widening the fastest in his "State of the Industry" report on casual games (pg. 9). From Microsoft's Xbox 360 Live Arcade, all the way to major PC portals such as Yahoo! Games and notable casual game creators and publishers, such as PopCap and PlayFirst, Hyman looks at where the casual game is positioned and where it's headed, as the market is predicted to skyrocket over the next few years.

The casual games market isn't the only one challenging how we define "player," "gamer," or even "game." The Independent Games Festival, awarded each March at GDC, recognizes independent developers and their creations, which continue to experiment and innovate in delightful ways. *Game Developer* applauds the finalists in a mini showcase in our news section (pg. 5).

RECALCITRANT CANINE

We're also very proud to feature an article by Richard Lemarchand, game director at Naughty Dog (CRASH BANDICOOT and JAK & DAXTER) who has written an eloquent piece on how his team made JAK X: COMBAT RACING in just 10 months, from full production start to gold master. Given the genre switch from action platform to combat racing and the short timescale, Lemarchand lays out for fellow developers the production methods that allowed them to complete the title so quickly to significant success without using time travel—or so he claims.

FEE, FI, FO, FUM

Of course, that's not all. Among the other highlights is another casual/indie game, NinjaBee's OUTPOST KALOKI X for Xbox 360 Live Arcade, showing off in the A Thousand Words art showcase, and a host of regular columnists making themselves known on some intriguing topics.

Programmer columnist Mick West writes this month on multi-core processors. Artist Steve Theodore covers how to delve smartly into art tool evaluation versions. Plus, Noah Falstein's design page looks at "the hunter and the hunted," and guest columnist Jesse Harlin of LucasArts provides a special discourse on Q/A practices for audio—a great subject that's often underplayed in even large studios.

DS: DOMINATION SITUATION?

For one reason or another, a number of my recent editorials have mentioned Nintendo in some fashion, and this one is no exception. The Nintendo DS's Christmas handheld domination of the Japanese hardware and software charts (Heads Up Display, pg. 4) is cause for both remark and fascination. For those who haven't been gazing glassy-eyed at Media Create's Japanese charts recently, Nintendo's conquering of the sales market has been practically unprecedented.

But here's the big question: Could the more extreme market widening that the Nintendo DS has prompted in Japan ever happen in the West, where currently, the DS and PSP are much closer to being evenly matched? To a certain extent, a greater market was reached for NINTENDOGS, but as the BRAIN TRAINING games start launching in the U.S. and Europe over the next few months, we'll find out whether Nintendo's world domination plans really span the entire globe.

S!

Simon Carless, editor

gamedeveloper
www.gdmag.com
CMP Media, 600 Harrison St., 6th Fl., San Francisco, CA 94107 t: 415.947.6000 f: 415.947.6090

EDITORIAL
EDITOR
Simon Carless *scarless@gdmag.com*
MANAGING EDITOR
Jill Duffy *jduffy@gdmag.com*
ASSOCIATE EDITOR
Brandon Sheffield *bsheffield@gdmag.com*
ART DIRECTOR
Cliff Scorso cscorso@gdmag.com
CONTRIBUTING EDITORS
Alexander Brandon abrandon@gdmag.com
Noah Falstein nfalstein@gdmag.com
Steve Theodore stheodore@gdmag.com
Mick West mwest@gdmag.com
ADVISORY BOARD
Hal Barwood Designer-at-Large
Ellen Guon Beeman Monolith
Andy Gavin Naughty Dog
Joby Otero Luxoflux
Dave Pottinger Ensemble Studios
George Sanger Big Fat Inc.
Harvey Smith Midway
Paul Steed Microsoft

ADVERTISING SALES
NATIONAL SALES MANAGER
Afton Thatcher e: *athatcher@cmp.com* t: 415.947.6217
SENIOR ACCOUNT MANAGER, EAST COAST, EUROPE & EASTERN CANADA
Ayrien Machiran e: *amachiran@cmp.com* t: 415.947.6224
ACCOUNT MANAGER, NO. CALIF., NORTHWEST, ASIA & WESTERN CANADA
Nick Geist e: *ngeist@cmp.com* t: 415.947.6223
ACCOUNT MANAGER, GLOBAL RECRUITMENT/EDUCATION & TEXAS
Aaron Murawski e: *amurawski@cmp.com* t: 415.947.6227

ADVERTISING PRODUCTION
ADVERTISING PRODUCTION COORDINATOR Kevin Chanel
REPRINTS Julie Rapp e: *jarapp@cmp.com* t: 510.834.4752

GAME GROUP MARKETING
DIRECTOR OF BUSINESS STRATEGY Michele Maguire
DIRECTOR OF MARKETING Tara C. Gibb

CMP GAME GROUP
VP, GROUP PUBLISHER APPLIED TECHNOLOGIES Philip Chapnick
CONFERENCE DIRECTOR, GDC Jamil Moledina
SENIOR CONFERENCE MANAGER, GDC Meggan Scavio
EXECUTIVE WEB PRODUCER Peter Leahy
EDITOR, GAMASUTRA.COM Simon Carless
FEATURES EDITOR, GAMASUTRA.COM Quang Hong

CIRCULATION
CIRCULATION DIRECTOR Kevin Regan e: *kregan@cmp.com*
CIRCULATION ASSISTANT MANAGER Jessica Ward e: *jward@cmp.com*
CIRCULATION COORDINATOR Miguel Mendiolaza e: *mmendiolaza@cmp.com*
CIRCULATION ASSISTANT Michael Campbell e: *mcampbell@cmp.com*
CIRCULATION ASSISTANT Andrea Abidor e: *aabidor@cmp.com*

SUBSCRIPTION SERVICES
FOR INFORMATION, ORDER QUESTIONS, AND ADDRESS CHANGES
t: 800.250.2429 f: 847.763.9606 e: *gamedeveloper@halldata.com*

INTERNATIONAL LICENSING INFORMATION
Mario Salinas
t: 650.513.4234 f: 650.513.4482 e: *msalinas@cmp.com*

CMP MEDIA MANAGEMENT
PRESIDENT & CEO Steve Weitzner
EXECUTIVE VP & CFO John Day
EXECUTIVE VP, CORPORATE SALES & MARKETING Jeff Patterson
SENIOR VP, AUDIENCE MARKETING AND DEVELOPMENT Bill Amstutz
SENIOR VP, INTERNET BUSINESS Mike Azzara
SENIOR VP, CMP INTEGRATED MARKETING SOLUTIONS Joseph Braue
SENIOR VP & GENERAL COUNSEL Sandra Grayson
SENIOR VP, CORPORATE SALES Anne Marie Miller
SENIOR VP, MANUFACTURING Marie Myers
SENIOR VP, COMMUNICATIONS Alexandra Raine
SENIOR VP, CORPORATE MARKETING Kate Spellman
VP, AUDIENCE DEVELOPMENT Michael Zane
PRESIDENT, CHANNEL GROUP Robert Faletra
PRESIDENT, CMP ENTERTAINMENT MEDIA Tony Keefe
PRESIDENT, CMP HEALTHCARE MEDIA Vicki Masseria
SENIOR VP, GROUP DIRECTOR ELECTRONICS & SOFTWARE GROUPS Paul Miller
SENIOR VP, ENTERPRISE GROUP Fritz Nelson
SENIOR VP, GROUP DIRECTOR COMMUNICATIONS GROUP Stephen Saunders

Game Developer is BPA approved
CMP United Business Media

MICROSOFT @ GDC 2006

EXPONENTIAL POTENTIAL

Microsoft
Game Developer Day

Tuesday, March 21
10:00 AM – 6:00 PM

Microsoft
Booth 416

Wednesday, March 22
Thursday, March 23
Friday, March 24

Microsoft
Sponsored Sessions

Wednesday, March 22
Thursday, March 23
Friday, March 24

Microsoft
Hands-On Labs

Wednesday, March 22
Thursday, March 23
Friday, March 24

www.gdconf.com

Microsoft® DIRECTX®

™ XBOX 360™

Microsoft XNA™

GOT NEWS? SEND US THE BIG SCOOP AT *EDITORS@GDMAG.COM*

MOBILE DEVELOPMENT HITS CES

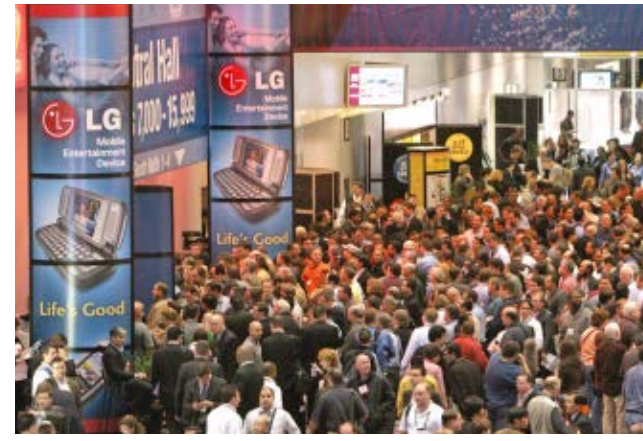
OTHER INDUSTRY PRESENCE FLAGGING AT MASSIVE CONSUMER EXPO

ONCE HAILED AS THE WORLD'S PREMIERE showcase for video games, industry presence at the Consumer Electronics Show (CES) has been on a sharp decline since the advent of a game-exclusive show (E3) in 1995.

Though it may no longer be the "World's Fair" of game innovation, as game journalist Bill Kunkel once described it, CES 2006 did manage to host a number of welcome surprises, most notable of which, apart from the Xbox 360 HD-DVD add-on, was a handful of development-centric discussions via Digital Hollywood's "Game Power" series of panels. Popular discussions and themes this year included the rising cost of game development, the issue of cross-platform convergence, and the continued growth of both the casual and mobile game sectors.

"We see amazing prototypes, with great and innovative game design, but we ultimately come to the decision that it won't sell, and we reject them," said Jay Cohen, vice president of publishing for Ubisoft, in a panel discussion on 2006 platforms. "And this can't go on. We want to bring these games out. We can't keep making these $50 million blockbuster productions." Cohen conceded that Microsoft's Xbox Live Arcade for Xbox 360 might create a solid market for creative, low-budget games.

With the Xbox 360 streaming audio and video from your PC to your television, and with the obvious popularity of cell phone gaming, convergence among devices is a hot issue in the electronics space, and the cause of more than a few headaches among game developers looking to produce games for more than one platform.



The Consumer Electronics Show, held January 5–8 in Las Vegas, drew a crowd of more than 150,000.

"We published an NFL game for mobile, and to make it compatible with as many units as we could, we had to produce three different builds by three different developers," said Jamdat Mobile vice president of sales and marketing Minard Hamilton in another Digital Hollywood panel, "New Opportunities in Gaming."

In that same session, Motorola's Jason Rubinstein discussed the enormous audience for the mobile games market. "There are 300 million actively used, game-capable phones in the world," he said. "There are, what, 12 to 13 million PSPs sold? The average price of a mobile game is around $4.55, which is about the cost of renting a movie, and PSP games are $40 to $50. So while [mobile games have] a very different user base, it's also a very big one."

—Frank Cifaldi

CALENDAR

Games for Women, Games by Women
Fort Mason Center
San Francisco
February 18, 2006
Price: $30–$45
www.womeningamesinternational.org

Video Game Financing Forum: Playing for Profit
The Helmsley Parklane Hotel
New York
February 23–24, 2006
Price: $1,795
http://insightinfo.com

Intel Developer Forum
Moscone Center West
San Francisco
March 7–9, 2006
Price: $595–$1,695
www.intel.com/idf

Game Developers Conference
San Jose Convention Center
San Jose, Calif.
March 20–24, 2006
Price: varies, see web site
www.gdconf.com

Nintendo DS Dominates Japanese Market

THIS HOLIDAY SEASON IN JAPAN, IT WASN'T THE Xbox 360 that sold out across the nation—it was the little handheld that could, the Nintendo DS. The dual screen console does indeed seem to be reaching new markets in Japan, as part of Satoru Iwata's plan to broaden the gaming sphere beyond the hardcore, a strategy that he outlined at the Tokyo Game Show last September.

Due in part to the compelling, yet broad nature of its software, the DS has sold more than 5 million units in Japan its first 13 months on the market, the fastest ever for any console in the nation. The Game Boy Advance took 14 months to reach the 5 million milestone, and the PlayStation 2 took 17 months.

In the lead-up to Christmas, the DS was consistently selling near to a half million copies per week, reaching sold out status by the new year, and outselling the PSP by 400 percent in that period. A formal apology was issued by Nintendo, stating that the company would restock as quickly as possible, but as of press time it was still almost impossible to find a DS in Japan, with prices marking up as much as twice the original value, in a phenomenon similar to the Xbox 360 shortage in America and the U.K.

The runaway success of the DS in Japan is due in large part to its original, cleverly differentiating software, with four games already having passed the one million sales mark in the region: NINTENDOGS, ANIMAL CROSSING: WILD WORLD, and the brain-training titles BRAIN AGE and BRAIN FLEX. The DS has also enjoyed healthy success in the West, with three games, NINTENDOGS, MARIO KART DS, and MARIO 64 DS having passed the million mark in America. While the Western market is certainly also a boon for the console, the PSP is much stronger in the occidental territories, by most estimates neck-and-neck with the DS.

—Brandon Sheffield

2006 INDEPENDENT GAMES FESTIVAL

THE ORGANIZERS OF THE INDEPENDENT GAMES Festival (run by the CMP Game Group, which also publishes *Game Developer*) have announced the finalists for the main 2006 IGF competition. Winners will be revealed at the 2006 Game Developers Conference in San Jose in March.

Following a record total of 118 entries, competition was extremely fierce, but the 40 IGF judges have singled out their picks for outstanding indie titles. This year's revitalized judge line-up includes a cross-section of game professionals, from developers at major studios (Nihilistic, Vicarious Visions, Shiny, Criterion/EA, Activision, Big Huge Games), through journalists from major indie game sites such as TIGSource, GameTunnel, and DIYGames, all the way to indie developers behind former IGF-prize winning titles such as GISH and N.

The finalists for this year's $20,000 Seumas McNally Grand Prize (pictured here) include Introversion's cult action-strategy title DARWINIA, Ankama's French strategy-RPG MMO DOFUS, Grubby Games' fiendish puzzle platform game PROFESSOR FIZZWIZZLE, Digital Eel's innovative "short" space exploration title WEIRD WORLDS: RETURN TO INFINITE SPACE, and Pocketwatch Games' ecosystem-building title WILDLIFE TYCOON: VENTURE AFRICA.

Also notable are the finalists for the Innovation In Game Design award, which include the afore-mentioned DARWINIA alongside block-stacking fighter RUMBLE BOX, single-button game STRANGE ATTRACTORS, time-bending platform title BRAID, and story-led adventure THE WITCH'S YARN. In addition, the new Best Web Browser Game finalists comprise Looney Tunes-esque Shockwave title DODGE THAT ANVIL, chemical puzzle game MOLECULOUS, and cheeky bullying brawler DAD 'N ME.

The remaining finalists are as follows. Technical Excellence: SAINTS & SINNERS BOWLING, TRIBAL TROUBLE, TUBE TWIST, DARWINIA, CRAZY BALL. Innovation In Visual Art: DOFUS, DARWINIA, PUTT NUTZ, GLOW WORM, THOMAS AND THE MAGICAL WORDS. Innovation In Audio: PROFESSOR FIZZWIZZLE, SAINTS & SINNERS BOWLING, DODGE THAT ANVIL, GLOW WORM, WEIRD WORLDS: RETURN TO INFINITE SPACE.

The main IGF competition, which has awarded prizes in the past to WIK & THE FABLE OF SOULS, ALIEN HOMINID, OASIS, and GISH, sports a total prize pool of $35,000 this year, including a biggest-ever $20,000 Seumas McNally Grand Prize for the best independent game, as well as individual $2,500 prizes for Innovation In Visual Art, Innovation In Audio, Innovation In Game Design, Technical Excellence, Best Web Browser Game, and an Audience Award, for which all finalists will be eligible. IGF Mod and Student Showcase finalists have also been announced (see www.igf.com).

—Simon Carless

SEUMAS MCNALLY GRAND PRIZE FINALISTS

Digital Eel's WEIRD WORLDS: RETURN TO INFINITE SPACE is an original PC "short game" that takes the player on a strategy-clever trip through black holes, bizarre alien planets, and frenzied ship battles in the space of a few minutes.

Pocketwatch Games' WILDLIFE TYCOON: VENTURE AFRICA is an intriguing eco-system sim title based in the African Serengeti, in which players balance food and water needs, breed animals, and create wildlife nirvana.

Introversion Software's DARWINIA is a stylish, almost retro-themed strategy action game developed by self-avowed U.K. "bedroom programmers," and was recently picked up for distribution on Valve's Steam service.

Ankama's DOFUS is a unique Flash-powered indie MMO that takes the popular strategy-RPG concept and spins it into a delightfully cartoon-styled jaunt in search of magic dragon eggs.

Grubby Games' PROFESSOR FIZZWIZZLE is a cunning 2D puzzle-action game that comes complete with classic THE INCREDIBLE MACHINE-style conundrums, over 230 levels, and a fully featured level editor to boot.

AUTODESK 3DS MAX 8

BY RONNIE ASHLOCK

JUST ABOUT EVERY NEW FEATURE OF Autodesk 3ds Max 8 software, the latest release of Autodesk's long-standing 3D modeling, texturing, animation, and rendering package, will seriously assist artists in getting their jobs done faster and at a higher level of quality. 3ds Max's reputation as a game industry standard is well-earned and 3ds Max 8 continues that tradition with new or improved intuitive and truly useful tools.

Don't expect any new interface changes, at least immediately obvious ones. 3ds Max 8 looks virtually identical to the past few releases—even to the point of having the same icon as Max 7. In fact, Max 8 really doesn't aesthetically differentiate itself from its predecessors in any meaningful way. Rather, it extrapolates upon the tried and true interface artists have come to know in their sleep.

3ds Max has never boasted a sexy interface, but the nuts-and-bolts practicality of its UI needs no such enhancements. An artist familiar with the program will feel right at home and can quickly pick up the mouse and get right to work. In addition, as with 3ds Max 7, existing proprietary plug-ins tested for this review worked with no problems. I loaded a few propriety CgFX shaders and they worked without a hitch.

LEGACY LETHARGY

Two problems with hanging on to the existing legacy are occasional inexplicable crashes and somewhat underwhelming performance with large files. First, 3ds Max 8 still crashes at the weirdest moments, though the number of crashes was tolerably low. I suffered only one crash over a period of two weeks of steady, demanding work with the program. It happened when I tried to delete three edges on a very simple plane—it crashed after I hit "Remove" in the interactive dialogue. I restarted the program and performed the same operation with no problems.

Second, large scenes have given—and still give—3ds Max problems. I imported a one million-poly model character reference mesh, made in ZBrush 2, into 3ds Max 8 to use as a reference object for a normal map, diffuse color map, and ambient occlusion map burn using the software package's Render to Texture utility. The program chugged into frustrating slowness despite the fairly resilient system I was working on (a P4 3.5GHz with a GeForce 6800 Ultra GT 256MB of RAM).

Extracting normal maps from high-frequency models for use on lower-poly, in-game models is standard operating procedure when making art for most current game engines. This is a process many artists use every day. Resorting to workarounds, such as degraded high-frequency models to a lower subdivision level or breaking them into chunks so the models can be pushed through the texture extraction pipeline, makes XSI's Gigapoly core and tolerance for large scene files especially appealing. Autodesk needs to seriously address this deficiency in the next release.

WHAT YOU MAY HAVE MISSED IN 7.5

Those not on subscription with Autodesk likely have not used the half-step upgrade 3ds Max 7.5, and thus haven't had a chance to check out the Cloth, Hair, and Fur tools. (If you've already used these tools, you're probably less enamored with them than I am.)

Prepare to be impressed—both by how easy the tools are to use and how good the results are. Cinematic artists will find these features very beneficial, though artists generating real-time content may not actually use them.

It takes some planning and practice to get skillful with Cloth, but the resulting clothing works great and responds to dynamics exceptionally well. Feedback with Hair and Fur is quick (styling the hair will likely be a guilty pleasure for a lot of artists as they spend inordinate amounts of time grooming their model).

I loaded a hair preset and used it as basis for the scruff I wanted on one of my creature models, and within minutes I had it modified to my satisfaction. Renders of Hair and Fur look absolutely amazing using either the default scanline renderer or the included Mental Ray 3.4 rendering engine. Again, this might not have tons of real-time use, but for sprites or cinematics, I think most people will really appreciate the value of these tools.

ENWRAPTURE

Of the newest features of this release, the one most immediately beneficial to character, environment, or prop artists is the Pelt Mapping functionality found in the improved UV Unwrap tools. This one feature easily makes the upgrade worth it.

With a few short clicks, Pelt Mapping allows artists to quickly "skin the cat" of UV coordinates on a model. By using seam edges, an artist can lay out how a mesh's UVs will be flayed onto a grid. I took a quadruped creature I had already mapped traditionally and remapped it using the Pelt Mapping tools. I easily cut the time in half and had better results. UV Relax tools have also been expanded to give artists the choice of relaxing UVs three ways: by center (the original way in previous versions), by face angles, or by edge angles, which gives the best result in my opinion.

Modeling also gains some beneficial improvements. The Bridge Edge tool allows artists using either an Editable Poly object or the mesh with an Edit Poly modifier applied to it to create a bridge between selected edges rather than limiting the

3DS MAX 8

STATS

AUTODESK, INC.
111 McInnis Pkwy., San Rafael, CA 94903, 415-507-5000

PRICE
Suggested retail price: $3,495; the subscription price is an additional $440.

SYSTEM REQUIREMENTS:
HARDWARE
- Intel Pentium III or AMD processor, 500 MHz or higher (Xeon or dual AMD Athlon or Opteron 32-bit system recommended)
- 512MB RAM
- 500MB swap space
- Graphics card supporting 1,024x768 16-bit color with 64MB RAM (OpenGL and Direct3D hardware acceleration supported; 3D graphics accelerator 1,280x1,024 32-bit color with 256MB RAM recommended)
- Microsoft Windows-compliant pointing device (optimized for Microsoft IntelliMouse)
- DVD-ROM drive

SOFTWARE
- Microsoft Windows XP Professional SP2 (recommended), Windows XP Home Edition SP2, or Windows 2000 SP4.
- Microsoft Internet Explorer 6
- DirectX 9.0c (required), OpenGL (optional)

PROS
1. Pelt Mapping is a joy to use and will cut down on the time it takes to UV a model.
2. Changes to both biped and non-biped animation rigs are welcome.
3. New motion editing tools and file formats will make life easier for technical directors and animators.

CONS
1. Still chokes on large scene files.
2. Some changes, while welcome, are a little overdue.
3. Occasional odd crashes.

OUR RATING SYSTEM :

EXCEPTIONAL | GREAT | FAIR | POOR | UNFORTUNATE

artist to just using the border of an editable poly—a great timesaver for making edits to either a character or a mesh. One slick feature added to the Editable Poly modeling tools is Clean Remove. By holding down the Control key when using the Remove tool in edge mode, related vertices are also deleted. Nice.

I quickly became addicted to working this way and found myself frustrated when I switched to another 3D package and it didn't have this feature. The ability to retain selection borders when switching to different sub-object selections (say, from edge to vertex mode) by holding the Shift key is also very useful.

The new Sweep Modifier is also a handy new tool environment artists will likely embrace. It is quicker and easier to use than the loft tools for creating trim geometry for objects like walls. Using the Sweep tool made me forget about the cumbersome process of lofting along a spline to get a similar result. While the default sections for use with Sweep are comprehensive, it's easy to create a custom profile using the spline tool and to use it instead of a default section.

Another new Editable Poly feature is the ability to punch holes using the Interactive Chamfer Dialogue. I can see how this will make mechanical modeling, both for characters and environments, more intuitive. Overall, the new modeling tools are definitely a real boost to productivity.

FUN WITH BIPEDS

Character Studio, rolled into the 3ds Max core since version 7, gains many new features, notably improvements to the Biped. Biped now has twists on all limbs (twist was previously only available to the forearm bones). These twists are handy for simulating muscle movement and it's nice to have them available for all the limbs. When I put them on a biped and skinned on a moderately dense mesh (8,196 quads), the resultant skin and twists worked really well. The mesh deformation was much cleaner than on a mesh without the twists. Also helpful is the ability to extend the amount of neck, ponytail, and tail links. Now a biped can have up to 25 segments each.

Skinning is also improved with the addition of the ability to weight all vertices of a mesh at the time you bind it to a skeleton, which results in never having a vert not applied to a bone at the time of skinning. And the default weighting was pretty good. I applied a skin using the default bone assignment to a fairly standard creature—a cave trollish-character—using a biped. I still had to do considerable tweaking, but over all it did save me some time.

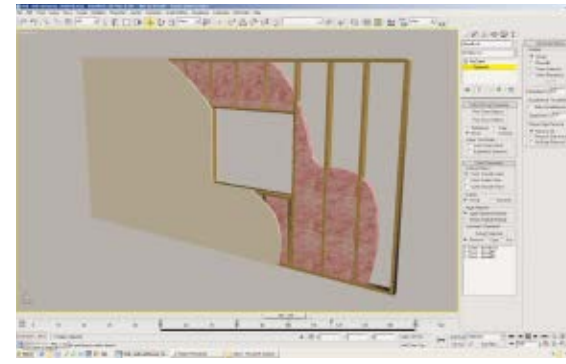
Animators should rejoice, as Biped now supports Euler rotations. Switching between Quaternion and Euler is simple (just click a button), accurate, and reversible, giving you the best of both worlds. Also of use to animators, the new XAF file format allows artists to save motion from one rig and quickly apply it to a completely different rig with just a few clicks using the motion retargeting interface.

Animators using Biped have long enjoyed the ability to move animation data between different characters, but the data was basically always shared with one rig, the Biped. With motion retargeting, saving animation data off of one rig in the XAF format is now possible, although it will take a little trial and error to get the hang of the process. Once you get it, though, you will wonder how you ever lived without it.

I was able to transfer a fairly complicated animation from a dinosaur onto a very simple proxy rig (with no constraints) in a few minutes, and the motion came across perfectly. The tool has enough flexibility to be appreciated (and expanded because XAFs are open XML files) by technical directors, but even non-technical types will, once they get the hang of it, find this tool invaluable. Motion Mixer gets an overhaul by allowing the same functionality available to Bipeds for virtually any 3ds Max object now. This is a very powerful and valuable upgrade. The ease of blending animation tracks onto a non-Biped rig is long overdue and greatly appreciated.

ASSET TRACKER

Keeping track of files has always been one of 3ds Max's weakest features. Autodesk addressed this problem with Asset Tracker and Vault. Essentially a resource browser, Asset Tracker is good for displaying, quickly and easily, where all the elements used in a scene (Xrefs, image maps and other dependencies) are located. I liked being able to select an object in the Asset Tree and then with one right-click of the mouse, to be able to set the object's path, locate it in Explorer, or to view it, if it was an image. This will save a lot of time over the life of a project as scenes inevitably grow more complex and remembering what each component actually does and what it looks like will become more difficult.



This 3D model of a dry wall assembly can be illustrated and animated with multiple splines using Pro Cutter to create a cut-away simulation.

Vault is the second part of the asset tracking solution provided by Autodesk. Vault is likely of interest to studios not currently using some form of source control, such as Alienbrain or Perforce, since it comes bundled for free with version 8. The Asset Tracker works either by itself or in conjunction with Vault and I really liked it. The files for both Vault client and server are included on the installation DVD.

UPGRADE A BARGAIN

Studios already on subscription have probably already received Cloth and Hair and Fur with 3ds Max 7.5, and have likely already upgraded to 8. Those not availing themselves of subscription might wonder if this new version is worth the money. Given the robustness of just a few features, the upgrade is definitely a bargain.

Pelt Mapping is elegant and extremely useful. It actually makes the chore of laying out UVs enjoyable. Biped has continued to grow in flexibility and features, and the new XAF format combined with the Motion Retargeting and the expanded Motion Mixer open 3ds Max up to animators as never before. The new asset tools are also solid and production-ready. The Max faithful do indeed have reason to rejoice.

RONNIE ASHLOCK *has more than five years experience making video games. He currently works for Sony Online Entertainment's Seattle Studio, working on an unannounced next-generation game. Email him at* ***rashlock@gdmag.com.***

CONFERENCE MAY 9-11 EXPOSITION MAY 10-12 L.A. CONVENTION CENTER

Three Days That Could Change Your Year.

WHERE BUSINESS GETS FUN

E3 expo 2006

One place. One entire industry waiting for you. One dynamic educational program that will guide you toward the most promising opportunities within the interactive entertainment industry today. It's the E3 2006 Conference Program.

Industry executives and renowned creative talent will share their knowledge and views of the future, and supply you with tips and strategies to perfect your business plans and keep you one step ahead of the competition. Whether you're a designer, manager or strategist, there are more than 30 sessions covering a wide variety of topics specifically designed to meet your needs and interests.

And that's not all. The exhibit floor is where you will see all the newest computer and video game products and technologies that the industry has to offer.

Register today for the E3 Conference Program.
Three days that could change your whole year.

REGISTER ONLINE WWW.E3EXPO.COM

E3 is a trade event and only qualified industry professionals may attend. No one under 18 will be admitted, including infants.
Visit www.e3expo.com for registration guidelines.

paul hyman

PHOTO: DEREK JONES AND ELENA BUETLER

STATE OF THE INDUSTRY: CASUAL GAMES

CASUAL GAMES? POPULAR GAMES? SMALL-FILE GAMES?

Coffee-break games? Web games? Even the folks who develop, publish, and sell them can't agree on what to call them. But, to paraphrase Justice Potter Stewart, who was unable to provide a definition of pornography, "I know 'em when I see 'em."

However, even categorizing the tiny, addictive, downloadable games that are the mainstay of game portals and mobile gaming may become more difficult if today's trends continue—the games are growing in size, becoming more content-rich, costing more to build, and some are being priced as high as $30, even more than so-called "value" or "bargain bin" console games.

Indeed, there is no longer anything casual about the casual games market, where expansion onto new platforms is becoming a very lucrative business for creative deal-makers.

Seattle-based PopCap Games, generally considered one of the leading developers in the space, is currently signing on to provide content for in-flight airline entertainment and interactive TV systems. There are even slot machine adaptations in the works.

"As more and more devices become enabled to play games, the most successful games on these devices are often casual games ... and that's because the more traditional or hardcore games can't find a big enough audience there," says James Gwertzman, director of business development at PopCap. "Casual games have such a broad reach and universal appeal that they can go almost anywhere, sort of like water dripping down through cracks."

Gwertzman says he's very bullish on the market, which he expects will continue to grow very rapidly. And why shouldn't he? A peek at the top 10 games list on most mobile phone services reveals that practically every title is a casual game. In fact, one of the more popular titles is BEJEWELED 2, PopCap's latest iteration of the flagship game that started the match-three genre and which has sold over six million copies.

Recent research by San Diego-based DFC Intelligence states that North American casual gamers spent $241 million on downloadable games in 2005 and are expected to spend as much as $1.7 billion by 2009. According to Alexis Madrigal of DFC Intelligence, "The casual game market definitely has serious growth potential. Today, the top casual game companies generate

PAUL 'THE GAME MASTER' HYMAN *was the editor-in-chief of CMP Media's* GamePower *and currently writes a weekly column on the video games industry for* The Hollywood Reporter. *He's covered gaming for over a dozen years. Email him at* ***phyman@gdmag.com.***

CASUAL GAMES

$10 million to $20 million in annual revenue, but in a few years, we could see some companies make $100 million or more from casual games."

IT STARTS WITH THE DEVELOPER

The PopCap web site (www.popcap.com) represents how the typical casual games business model works. A selection of 25 free web-based games—which are monetized by advertising—act as teasers, encouraging players to step up to what PopCap calls the deluxe versions of their games. While some people never do anything but play the browser-based freebies—which contain only a few levels and sport reduced graphics and sounds, but can be played for an unlimited amount of time—they are encouraged to download the more complete game and try it for an hour or so. After that time period, the game ceases to function but gives the player the choice of revving it up again with a credit card charge of around $20.

"We're finally seeing the industry start to experiment a bit with different prices and different free-trial periods," says PopCap's Gwertzman, "but in most cases, that remains the basic model."

The typical industry-wide "conversion rate"—meaning the percentage of people who try the game and then decide to buy it—is anywhere from 0.1 to about 1.5 percent for an average game, or from 1.5 to 2.5 percent for the highest-quality games. PopCap averages 2 percent.

At that rate, it's clear that the casual gaming market depends on high volume. If after spending an average of several hundred thousand dollars to build a game, a developer never sees a penny from 98 percent of the users who try it, if only 2 percent of their audience shells out $20 to buy the "registered version" of the game, it's obvious that developers need to maximize the size of their audience, and maximize it majorly.

John Welch, president and CEO of the casual game publisher PlayFirst.

This can be done in a number of ways.

The first way, obviously, is to be known as a developer that makes great games so that people seek out your games instead of the competition's. Gwertzman estimates that, in its five-year history, PopCap has seen over 150 million downloads of the 25 games it has published, all of which have been profitable. And a "high percentage of them have been hit games," he says, defining a hit casual game as selling 100,000 units or more. He reports that, in 2004, PopCap held 15 percent of the casual games market.

And Gwertzman says he's more than willing to assist other developers who are ready to innovate.

"We've released our open-source PopCap engine (http://developer.PopCap.com), which is the exact same engine we use for our own games. We've made it free for any developer to download and use," he says. "There's no licensing fee connected with our PopCap Developer Program. All we ask is to give us credit in the game."

He explains that providing an open-source engine is the best way he knows to encourage small developers to be original and to remove the distraction of having to build their own engine.

"We just give them ours to start with," he says, "and let them focus on making great games so there's no need to reinvent the wheel."

PopCap's recipe for building a successful casual game, says Gwertzman, is to eliminate schedules and go through months of endless prototyping before serious production begins.

"One of the reasons we've made so few games is that most of our ideas end up on a shelf and aren't released," he explains. "Many people think that all you have to do is throw a bunch of

PopCap's BEJEWELED 2.

games at the wall and see what sticks. We think nothing can hurt this industry more than a flood of crappy games."

He describes the best PopCap games as "very, very relaxing and fun. People play our games as almost a diversion, not to compete with other people, not for an adrenaline rush. Many hardcore gamers look at casual games and they just don't get it. There's a mode in BEJEWELED 2 in which you can't die; there's no end-game criteria. But that's really not the point of the game ... which is that it's relaxing."

An alternative approach to the casual game space is exemplified by San Francisco-based casual publisher PlayFirst.

WHEN PUBLISHERS ASSIST

It was less than two years ago, in April 2004, when John Welch and his partner Brad Edelman decided it was time the casual games sector got its first full-service publisher: PlayFirst.

"Games were getting more expensive to create, distribution was becoming more complicated, there were opportunities to distribute under multiple platforms in different countries with localized versions, and the complexities around the business side of things—finance, marketing, sales, distribution, legal—were all intensifying," says Welch, the company's president and co-founder. "The small developer teams were ill-equipped to do all that, and it was taking away from the focus of what they're really good at and passionate about—building games."

Kenny Dinkin, PlayFirst vice president.

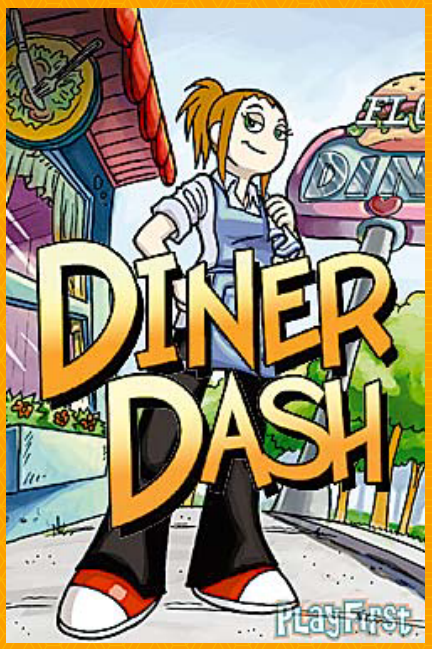
Specifically, Welch says, the budget for developing a single game has risen from around $50,000 or $75,000 a year ago to between $100,000 and $150,000 or more today. "That's because games like our best-selling DINER DASH have raised the bar in terms of innovation, depth, and animation." File size, too, has been expanding, with today's casual games straining at the 10MB ceiling.

Not only does PlayFirst finance the cost of the game's development and then manage the IP across multiple platforms, but most importantly, it specializes in signing deals with the various internet portals, such as Yahoo!, AOL, and MSN Games, to maximize the game's distribution.

"Think of it this way," says Welch, "a developer just needs one contract with PlayFirst, and then we handle all the distribution contracts and the servicing of the retailers."

Typically, a publisher will make the best deal it can with as many web sites and portals as it can, earning a percentage of the revenue when its clients' games are sold, and then sharing that revenue with the developers. Royalties can vary greatly depending on the deal.

"We're constantly working to expand the types of revenues we're getting for games, as well as creating very strong relationships with the bigger accounts so we can get more revenue than would otherwise come to an IP if someone just tossed it over the fence to the portal," says Kenny Dinkin, vice president and executive producer at PlayFirst.



GameLab's DINER DASH.

Dinkin manages PlayFirst's portfolio of six titles and is responsible for picking and choosing what sort of games the company will handle—a combination of science and art, he says.

"We have a lot of people over 40 playing our games, many of whom are female, and so we're trying to create a portfolio that meets the needs of that marketplace," Dinkin explains. "DINER DASH, for example, which features a character named Flo who quits her job to start a restaurant and then builds a restaurant empire, was really fresh when we first saw it. The developer, GameLab, did a brilliant job of blending a cute, aspirational theme with really compelling gameplay that one can play for hours on end. It was a great first product for us; it really spoke to the kind of portfolio we were interested in developing."

Whether a game is accepted or rejected, says Dinkin, depends entirely on his portfolio plan and whether the title fits into the categories he's looking for, perhaps a word game or a strategy game or an action puzzle. "We're looking for games where the mechanic is very accessible and meets the needs of the mass popular market. But, at the end of the day, a great, innovative idea that we think is going to succeed—something that stands out and isn't a [knock-off] title—will get our attention and our funding, regardless whether it fits into the categories we're seeking."

He warns against too much creativity, too. "If it's too unique or something the mass market will have trouble sinking its teeth into, then you run a big risk of not finding an audience, regardless which portal you're selling into."

However, Welch says he's always on the lookout for talent. "If you're a hotshot programmer or artist or producer or designer, if you are itching to move from being one person on a 100-person development team to being on a three-person team and have more control over what you're creating, if you have what it takes

CASUAL GAMES

to start your own company and approach PlayFirst with an idea, then 2006 will be a great year to go for it."

PORTALS MEAN SALES

Indeed, over at Yahoo! Games, one of the largest portals—with 23 million unique gamers visiting each month—picking the right games to offer is "part of the magic," says Thom Kozik, senior manager of business development. Yahoo! Games currently stocks just over 230 casual games and gets between 12 and 15 submissions from publishers and studios each week of which only two or three are added while a similar number are removed. The process is open to anyone who knocks on Yahoo!'s door.

"We pick what we think will be the best, play them, and choose the ones that are the most fun," he says. "It sounds simplistic, but we can't afford to alienate the customers in our storefront. They come to us to see the best games week in and week out, to try them, and maybe to buy them. If they find junk, they're going to go elsewhere. We could have a bigger catalog but chances are there would be more low-quality product in there."

WildTangent's FATE.

The trick, he says, is to find just the right games that will resonate with Yahoo! Games' audience, 80 percent of which is between the ages of 21 and 60. Forty-five percent is female.

Thom Kozik, senior manager of business development at Yahoo! Games.

"We offer a pretty wide shelf on our storefront," says Kozik. "Some of our audience likes to see similar kinds of games—like the three-in-a-row BEJEWELED variety—while others are looking for something new that they haven't played before."

Kozik admits that, except for a few exclusives, Yahoo! Games stocks most of the same games as everyone else but says the portal does better than most of the others in its ability to "personalize" its services.

"It comes down to merchandising a game that we know is probably one you're going to enjoy based on the other games you've played or purchased," Kozik explains. "We've had user ratings in our music area for a while and we just turned that on in the games area as well; gamers can provide ratings for the games they play giving us the ability to make suggestions based on what they liked or didn't like. For instance, if you just bought and liked BEJEWELED 2, we might offer you several other three-in-a-row games."

Yahoo! expands the gaming experience in other ways, too. Players can compete online against a friend in web-based casual games, they can check on their rankings in online tournaments that Yahoo! runs, and they can read reviews of casual games and then download them.

"We've found that offering an entire gaming experience under one umbrella, under one brand, has a lot of appeal, which is the reason we rack up over 7 billion minutes of online gameplay a month," says Kozik.

Indeed, Yahoo! Games is even more likely to pay more for a game that is capable of integrating with the rest of the Yahoo! network. Typically, the portal's deal with publishers and developers is based on a revenue share that depends on Yahoo!'s relationship with the game company and the game itself.

"Everything is negotiated," says Kozik, "but if the game reports point out to a central Yahoo! profile allowing gamers to see all their scores in a central location, if it integrates well into the Yahoo! overall experience, we can be more generous since it's worth more to us."

Recently, Kozik says he's seen the definition of a casual game starting to stretch to genres that are more complex than, say, simple puzzle games. For example, he describes WildTangent's FATE as a "lightweight role-playing-style game" that isn't necessarily what one would expect from a full-blown RPG like WORLD OF WARCRAFT. It is this sort of expanded casual game that Yahoo! Games is using to test the pricing waters by charging $30 for a registered version as opposed to the usual $20.

"We're using games like FATE as a way to broaden our offerings, to put a premium product on the shelf right next to the bargain products," he says. "We think casual games are shifting away from the one-size-fits-all strategy and we've found that the sales take-up has been very good because gamers see the premium games as having higher-quality and a longer gaming experience. I mean, FATE could easily keep you occupied for a month or more."

Another reason for charging more for "premium games," says Kozik, is that the studios and publishers are looking to recoup their larger investments in developing them. While simple three-in-a-row puzzle games can cost anywhere from $50,000 to

Business is busting at the seams for Orlando's digital media sector. Home to top-notch studios like Electronic Arts, specialized higher-ed training programs, and the world's largest concentration of simulation developers, it's no wonder companies around here are so animated.

CALL 888.TOP.CITY OR VISIT ORLANDOEDC.COM

Putting imagination to w rk®
ORLANDO

STATE OF THE INDUSTRY: CASUAL GAMES

Greg Canessa, group manager of Xbox Live Arcade.

$150,000, Kozik says he "wouldn't be surprised if creating FATE was closer to $1 million." Actually, FATE cost $250,000 to create, according to WildTangent CEO Alex St. John.

CASUAL GAMING FOR THE CONSOLE

Microsoft is hardly a newcomer to the casual game space. Its MSN Games portal carries about 120 titles and the audience for its puzzle games, which is the top category on the site, is a whopping 70 percent female.

"Casual gaming on the PC has grown from nothing to a multi-hundred-million-dollar business in a few short years," says Greg Canessa, group manager for Xbox Live Arcade. "It's projected to be $900 million by 2009, according to IDC."

Determined to transfer a chunk of that success to the video game console sector, Microsoft launched Xbox Live Arcade in November 2004 and quickly discovered that it had a hit on its hands.

"The conversion rates, from free trial to paid, were staggering," Canessa says. "The industry average for PC-downloadable games is under 1 percent, and we were getting a sustained average conversion rate across our 20 titles of almost 12 percent. Although we only had an install base of several hundred thousand users and despite the fact that Xbox Live Arcade wasn't optimal by any means from a user experience standpoint, even with all those barriers, we knew we were onto something."

Canessa is very candid about the complaints he heard from users. "Unless you had the disk from the starter kit inserted, you couldn't download a game, unlock it, or even play the downloaded games that were on your hard drive. And you needed to have a $50 Xbox Live subscription; there was no free level of service."

One year later, an upgraded version (Xbox 360 Live Arcade) launched concurrently with Microsoft's next-gen console and Canessa says it eliminates all those previous hurdles. The disk is gone and the service is fully integrated into the console and is available immediately for free. Canessa says that, through careful management of Live Arcade's portfolio, he is making a concerted effort to widen his audience.

"When you first launch new console hardware, the first five million units that you sell are really geared toward the hardcore gamer, the kind of person who waits in line in the snow for three days to pay $500 to be one of the first to get his hands on one," Canessa explains. "So we decided to launch with 70 to 80 percent of our portfolio appealing to a primarily hardcore audience. That doesn't mean we're doing HALO shooters. It means we're doing games that we've identified as having cross-over appeal, games that everyone—even hardcore gamers—will love."

In Canessa's opinion, that means the right mix of several popular genres. As a result, the 10 launch titles on Xbox 360 Live Arcade included three casual games (BEJEWELED 2, HEXIC HD, and ZUMA), five retro-type games (JOUST, GAUNTLET, GEOMETRY WARS RETRO EVOLVED, MUTANT STORM RELOADED, and SMASH TV), a strategy game (OUTPOST KALOKI X), and a bar game (BANKSHOT BILLIARDS 2).

"We're doing something that's really a lot broader than the web portals," Canessa explains. "We've got a much wider variety of genres than you find in the casual-game space."

He reports getting a dozen new submissions from studios and publishers each week, and he expects to have expanded his offerings from 10 last November to at least 35 this summer.

"Because we have fewer games than the PC portals, we need to keep tight control over our portfolio," he says. "We want to maintain that quality bar, we want to have a nice spread that appeals to our entire audience, and we need to avoid having [knock-off] titles. The idea is not to offer up, say four marble games, because that kind of redundancy doesn't help us, it doesn't help the developers, and it certainly doesn't help our users."

As the quality bar rises on casual games, the chance of creating something notable with a small team diminishes. And yet, there still remains a variety of opportunity in this expanding market.

PlayFirst's TRIJINX.

INDEPENDENT GAMES FESTIVAL

REWARDING INNOVATION IN INDEPENDENT GAMES

THE 8TH ANNUAL

INDEPENDENT GAMES FESTIVAL

MARCH 22-24, 2006
SAN JOSE, CA
WWW.IGF.COM

PLAY THE BEST INDIE GAMES OF THE YEAR!



PLATINUM SPONSOR:
[adult swim]

GOLD SPONSOR:
Microsoft

MODDING PAVILION SPONSOR:
EPIC GAMES

MEDIA SPONSORS:
GAMESPOT
cnet DOWNLOAD.COM

STUDENT SHOWCASE SPONSOR:
DIGIPEN INSTITUTE OF TECHNOLOGY

TOOLS CATEGORY SPONSOR:
ageia

CMP
United Business Media

VOTE ONLINE

FOR THE AUDIENCE AWARD— FIND OUT MORE AT WWW.IGF.COM

- Visit the IGF Pavilion at GDC
- Attend the IGF Awards ceremony Wednesday, March 22
- All GDC attendees are invited!

GameDevelopers Conference

MARCH 20-24, 2006
SAN JOSE, CALIFORNIA
WWW.IGF.COM

Register by February 15, 2006 and save up to 35%

Use priority code IGFMAXX when registering


GameDevelopers Conference
MARCH 20-24
SAN JOSE, CALIFORNIA
WHAT'S NEXT
GDC:06
www.gdconf.com
GAME DEVELOPERS CHOICE AWARDS
INDEPENDENT GAMES FESTIVAL
GDC MOBILE
SERIOUS GAMES SUMMIT
GAME CONNECTION
The next generation will be defined by the games you create. At GDC:06, learn lessons from next-gen, handheld, and current game development, and gain access to the people, technologies, and tools that will define *what's next* for our industry.
LEARN MORE AT WWW.GDCONF.COM.
REGISTER BY FEBRUARY 15, 2006 AND SAVE UP TO 35%!
USE PRIORITY CODE PAMAXX WHEN REGISTERING.
CMP
United Business Media

FROM SMART TO FINISH

JAK X: COMBAT RACING AND THE NAUGHTY DOG PRODUCTION METHOD

IN THE FALL OF 2004, WE AT NAUGHTY DOG INC. (CRASH BANDICOOT and JAK series of games) set out to create a game in a genre that was relatively new to us. And we had only 10 months—the shortest product cycle in the history of the JAK games—between the beginning of full production and gold master to do it.

This is the story of how we pulled it off, how Naughty Dog worked swiftly without compromising quality or resorting to human sacrifice.

INVITATION TO RACE

In order to be able to work as quickly as possible, we always start a project with a clear and simple summary of what we want to achieve. Collaborating with our producers at Sony, game director and Naughty Dog co-president Evan Wells defined three basic goals for the game.

First, we wanted to make a combat racing game using the physics-based vehicle gameplay from JAK 3, since it received a lot of positive feedback. Second, the game would be Naughty Dog's first online multiplayer game, a change we were all very excited about and had been anticipating for a long time. Finally, the game would have a strong single-player story mode with around 40 minutes of the movie-quality animation that the JAK games are known for.

Having a good plan and sticking to it is the best way to make a game efficiently, and working without wasting time or resources is the key to working fast.

CONTINUED ON PG 18

RICHARD LEMARCHAND *is a game director at Naughty Dog and was the lead game designer on JAK X: COMBAT RACING. His other credits include JAK 3, SOUL REAVER, and GEX. Email him at* ***rlemarchand@gdmag.com****.*

CONTINUED FROM PG 17

JAK X: COMBAT RACING character sketches drawn by Bob Rafei.

SHIFTING GEARS

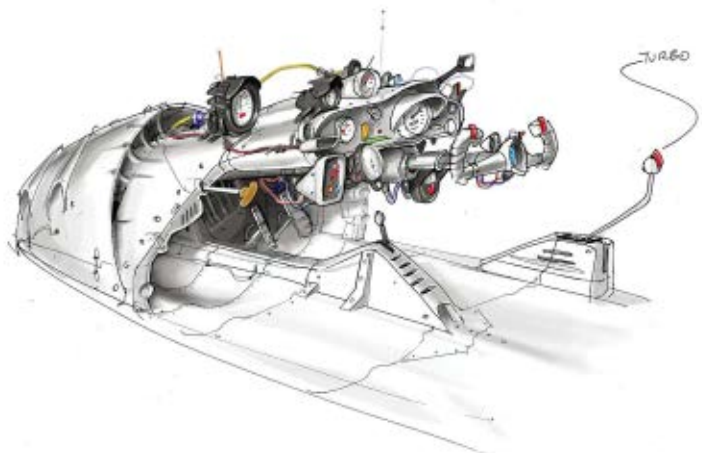
A new type of game demanded some new design and production techniques, but many of our methods stayed the same as those we used on JAK 3. Our development approaches are very technology-driven, and without deliberately following any particular dogma, many of our best practices run parallel to the software development philosophy known as agile development.

For example, on any project we always keep the game code running healthily and close to being turned into a finished package that a player could navigate and enjoy. We never leave the game broken or misbehaving for longer than an afternoon. Our frequent demo and focus test deadlines keep us focused on this goal and let us stay relatively stress-free in the face of what are sometimes weekly milestones to produce a new disc for E3 or a press event.

We develop concentrically—that is to say, we implement the fundamental mechanics of the game first, working to a good level of polish. Then we work outward through the secondary and tertiary mechanics, pushing upstream sensibly to implement more gameplay systems when there's a dependency. This lets us get a well-playing basic game together quickly and immediately provides a context in which we can evaluate new mechanics. If something isn't as much fun as we thought it would be (as was the case with some of the weapons we prototyped), it can easily be ditched without undermining other gameplay systems that might have depended on it.

This emphasis on working code rather than lengthy specifications, and treating change as an opportunity instead of a crisis, is characteristic of agile development. We consider agile development to be more of an attitude than a set methodology, and it's great for getting the very best out of smart, hard-working people. It also helps to keep the potentially stressful development process enjoyable and engaging, and if it's fun to make, then it's much more likely to be fun to play.

Another development method we used on both COMBAT RACING and our previous two projects was to adapt the engine of the previous game, giving us a big, time-saving leg-up in terms of functions and features.



JAK X is Naughty Dog's first game to offer online play.

This being our first online game, we developed the core of the online multiplayer game first—a new approach for us. Usually, we start with a mission partway through the single-player game. Getting a working version of the game up quickly allowed us to make discoveries that helped us plan better. Almost before we knew it, we were holding deathmatches across the network, which allowed us to see that our core concept was indeed a lot of fun, but that we'd need some HUD iconography to help players locate each other in the arenas.

Another new method that helped keep things moving quickly was to make a weekly build of the game from almost the beginning of development. Weekly builds could be seen as a new manifestation of our "always keep the game running" philosophy, and it paid dividends. Just a couple of months into development, we were able to play together online every Friday night from home, and that inevitably spurred in-depth conversations over the USB headsets about what (and who) ruled and sucked in the game.

Even though we had a game engine already, we needed to make considerable modifications to accommodate the new game style. Naughty Dog's programming director Christophe Balestra led the coding team as they tackled the new challenges of making objects networkable, meeting the changed spooling demands of the new style of game and handling the more developed vehicle physics and play features of JAK X.

Finally, to enable everyone to play the game as it grew, we immediately set up an interface shell that gave the developers easy access to the game's events.

TURBO BOOST FROM THE STARTING LINE

We knew that with such a short development cycle we had to hit the ground running, technologically speaking—and so we did. Around five months before the official November 2004 start date of the project, one of the lead programmers Ben Stragnell, an industry veteran and networking expert, began writing network code that would integrate into the JAK 3 code base.

We also prepared ourselves on the design front. We started thinking and talking about the game design of JAK X even before JAK 3 had gone gold in the late summer of 2004. We made sure to keep an exhaustive record of our initial ideas, since we've found that our first notions about a given gameplay subject are often some of our best.

Our preparations were completed when we duplicated the JAK 3 code and content base, quickly built some simple test tracks and arenas, and started driving around them immediately. This gave us a concrete mock-up of the game we were going to build, and we began to experiment, iterate, and most importantly play.

DESIGNING MAPS TO DRIVE BY

Naughty Dog's small group of game designers met almost every day for the first three weeks of the project, brainstorming for rarely more than two or three hours at a time, and then breaking to document and reflect on what was discussed. The product of this period was a concise 30-page design document.

The design discussed three things: top-level objectives for the project, the basis of the story, and key play features of the game. These included notes about the customizable cars and a macro-level spec of the resource economies we expected the game to have. We also kept several lists of ideas for locations, event types, and weapons, which ultimately provided nearly all the ideas we ended up using, as well as some extras. We planned from the start to build the Adventure Mode macro from a subset of the location/event matrix, depending on which combinations we thought were the most fun.

The design document was handed around to the team, and was updated a few times near the beginning of development; for the rest of the project we used a wiki to create living project documentation driven by need, linking it to schedules, asset lists, and other documents on the network that the team would find useful but might not find on their own.

Early on, we discussed the pros and cons of physics-based gameplay, and foresaw the tuning and balancing challenges we would have to meet. This let us approach a difficult job with our eyes open, which allowed us to dedicate the proper amount of tuning time to it.

Thanks to Naughty Dog's creative director and scriptwriter Dan Arey, our story planning saw a shift from a linear branching plot to a notion-based story arc with components that were fairly freely interchangeable. This approach to story became extremely useful as we constructed and reconstructed the Adventure Mode macro. Our phenomenal team of animators worked very hard to create the movies on a tight schedule and, as ever, achieved great results, including the most complex intro sequence that Naughty Dog has ever attempted.

Some other important design work that we tackled early on was to design the game flow and interface. Game flow in previous JAK games had been fairly simple. However, with an online multiplayer game, we knew we'd have to accommodate players getting online, creating games and browsing those of others, and personalizing and upgrading multiple vehicles, while having freedom of movement between all these actions and others. We knew we had to have a solid plan before we began to implement the game flow in earnest.

After careful analysis of some competitors' games, we created a huge flowchart that seemed to let players do everything they could want. Needless to say, we didn't get it exactly right the first time, but had to refine the design as we implemented it.

We also put a lot of effort into the design of the interface and spent part of November making simple screen mock-ups focused on functionality that we continually critiqued and improved, eventually building a prototype in Macromedia Flash. We decided to leverage our 3D strengths, and instead of the 2D interface we had initially planned, built a stylish and animated polygonal one. It took a lot of iteration and polish time, but we were happy with the results.

CONTINUED ON PG 21

Games of the Future

Game Connection @ GDC
20-21 March 2006 - San José - California

Game Connection is the premier event for innovative game developers to meet with top publishers in the game industry. The event is designed to be a highly valuable tool for developers to show off their talents to publishers who are looking to find the right project or the right studio to develop games based on intellectual property rights or licenses they own.

GAME CONNECTION

www.game-connection.com
Information: gconline@game-connection.com

Sponsored by GameDevelopers Conference

Alias | MotionBuilder

CONTINUED FROM PG 19

Finally, art director Bob Rafei (the man to whom JAK and CRASH BANDICOOT owe their stylish lines) and our awesome concept artists began creating the images that would lead the art direction for the game assets. Working art ideas out on paper can save a huge amount of time and effort (hence, money), and can inspire new thoughts about the game design. We made sure that we got off on the right foot with great concept designs for the vehicles (courtesy of a talented young transport design major named Hugo Martin) as well as game "furniture," such as power-ups and race banners.

As you can tell from the amount of work we applied to it, we value planning very highly as a key way to keep the project moving along at top speed.

WHERE THE RUBBER HITS THE ROAD

Naughty Dog has an in-house level layout guru, Hirokazu Yasuhara, one of three core members of the original SONIC THE HEDGEHOG team. Yasuhara first draws cartoons that illustrate the play mechanics for a level, and then simply uses a pencil, squared paper, and his visual imagination to draw a level map. A background artist then quickly makes a block mesh version of the level, which Yasuhara and the other designers can play test and fine-tune, while the artist simultaneously moves on to producing the finished background art.

Over the course of the last few JAK projects, this level layout process proved to be predictable and reliable in terms of the time it takes and the quality of the levels it produces. Nevertheless, we immediately performed a level layout test for JAK X, and by the end of the first month of development, we had our first level up and running.

At the same time, we created simple spreadsheet schedules for layout and art that kept the level design ahead of the artists implementing them. Since there was a wide variety of level sizes, we used a combination of experience and gut feeling to schedule the artists, based on an initial educated guess and working in collaboration with each background artist. Our artists are mostly very senior and are given a lot of responsibility for hitting their deadlines. Slipping is not an option, except in very special circumstances, which makes the team largely self-producing and keeps the bar of quality, organization, and professional conduct very high.

We entered full production at the beginning of November. By the winter break, we had completed three of the game's seven arenas and had built many of the levels comprising the various circuit tracks. We continued working on background art steadily through mid-July and were all but done by alpha.

A short way into production, we shifted toward spooling tracks that would dynamically load ahead of the player's progress through them. As a result (and as usual with the JAK games), we had to put a lot of serious thought and consideration into planning each level loading scheme, but in the end we got longer tracks with improved graphical variety, and hybrid tracks that would mix up different locations. The loading scheme also helped us get into and out of the menus with almost no loading times.

LAP TIMES AND POWER-UPS

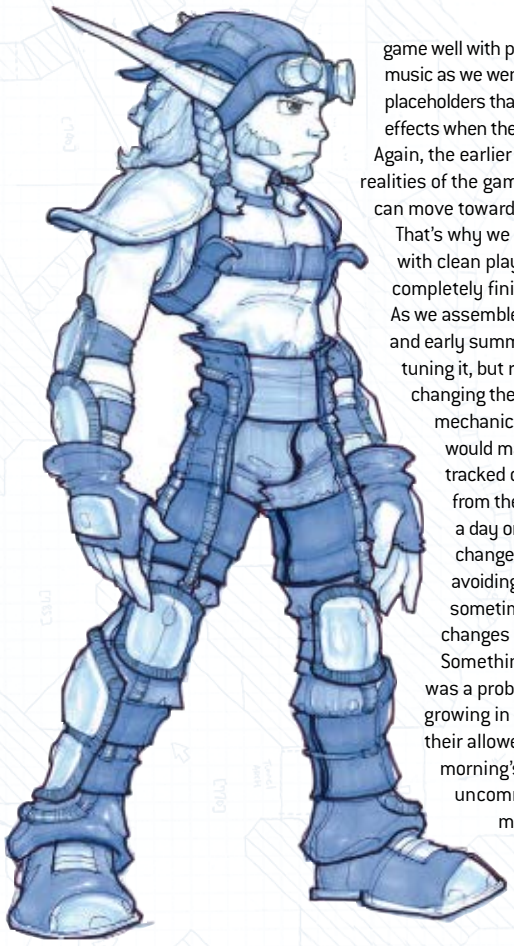
Working intensively with the programmers, artists, and audio crew, we began to implement the game promptly and rapidly, starting with the core mechanics and events and working out toward the more experimental types of gameplay. Game designers at Naughty Dog also act as producers, facilitating communication, clearing dependencies, iterating to raise the quality bar, and generally making a nuisance of ourselves. We like this method because it keeps the team lean, mean, and focused on the concrete goal of making the game, rather than getting bogged down by bureaucracy and deferred responsibility.

We always performed level implementation tasks (like placing power-ups and setting up signal planes) as soon as the levels became available in block mesh form to grow the game's footprint very rapidly and keep a good handle on performance issues. We would constantly play the game and try to provide timely and consistent feedback to the artists and programmers, who typically implemented any changes we asked for immediately. That way, we could keep our focus as we iterated quickly to a finished level of quality.

Whenever something wasn't quite coming together, a designer would sit down with the appropriate team member and hammer out a solution until we got it just right—Naughty Dog's version of extreme programming! GOAL (the proprietary LISP-like language that we write the JAK games in), along with our in-game editing tools, facilitates real-time changes and experimentation so we can reach optimally fun implementations of mechanics quickly and easily.

We put a small number of very experienced in-house playtesters on the project early, only about two months into production, which let us get some great feedback about the gameplay right out of the gate and smoothed the bug-fixing process as the first strands of the game were drawn together in early 2005. We worked continuously with our visual effects and audio staff to cover the

Level designs by Hirokazu Yasuhara are first done on paper, then mocked up by a background artist.



The JAK series' unique character design is part of what sets it apart from other games.

game well with particles, glows, sound effects, and music as we went along, occasionally using placeholders that could easily be switched out for final effects when there was a good reason for doing so. Again, the earlier we can make discoveries about the realities of the game we're making, the more quickly we can move toward a finished, great-playing game.

That's why we always value ASAP implementation with clean play and clear, if not necessarily completely finished, visuals.

As we assembled the game throughout the spring and early summer of 2005, we were continually tuning it, but rather than constantly and broadly changing the parameters underlying the game mechanics, we worked in a controlled way. We would make a small number of carefully tracked changes, gather feedback about them from the testers and team over the course of a day or two, and then make another batch of changes. We moved toward our goals while avoiding the churn and tail-chasing that can sometimes result from making too many changes in an uncontrolled way.

Something that occasionally slowed us down was a problem with the levels inexplicably growing in memory size overnight, exceeding their allowed maximums and spoiling that morning's disc with crashes. The problem isn't uncommon when a lot of people are making minor changes to the game, but because we didn't use version control software for the content, it was hard to track down and roll back the changes that had caused the growth.

Occasionally, we paused to remind ourselves of our initial goals, to make sure that the design hadn't drifted off course. Our game was designed to be interoperable with Ready at Dawn's DAXTER PSP (each game unlocking features in the other when connected together by a USB cable), and so we started cross-development with the other team early to avoid any unexpected problems.

Throughout development, we maintained lists of the secrets and bonus content we were planning, so that when the time came to add them, we moved very quickly. We also localized the game in real time using specially created tools and a lot of elbow grease, and shipped the game with all the languages on one disc.

By mid-March we had enough events, locations, and gameplay features to unveil JAK X: COMBAT RACING at a special press event. Even though the gameplay would continue to evolve and grow in the following months, our methods supported the making of a complete demo that played well enough to wow the assembled journalists and editors.

FINAL LAP

Well in advance of our alpha date in early July, we had expanded our in-house Q/A department to around 12 people who were technically savvy enough that we could teach them how to build and burn daily discs, taking considerable pressure off our midnight-oil-burning programmers.

The naturally scalable type of game that we were making let us evolve our best events through natural selection. We maxed out at about 15 events and whittled them down to the best 11 by beta in August. We were also able to cut a level fairly close to the end of development when the schedule demanded it, a great luxury for people used to working on character action games, whose crystalline, self-dependent structure usually creates headaches for people looking to prune levels.

To everyone's delight, JAK X ran very smoothly overall, but one unexpected snag we hit was that the full footprint of the game was much larger than we had anticipated, which led to some difficulties in getting everything tested thoroughly. Looking back, we should have developed a more explicit test plan and tracked its progress to avoid the scramble we had at the end to give every level and event the polish it deserved.

Starting as early as April 2005, we had begun to collect game metric data—the single best method for tuning a game. We held focus tests at Sony Computer Entertainment of America in Foster City and recorded information to the PlayStation 2 memory card about the unfolding game state of each test player, which we then compiled in spreadsheets and analyzed to see how players had progressed and to locate difficulty spikes.

We had a great online public beta in June that harnessed the power of PlayStation.com's message boards to generate invaluable player feedback, and we could always count on our extremely candid Q/A crew to tell us exactly what was fun and what was not. The metric capture sessions continued right up until the last weekend before beta, as we scoured our networks of friends for rookie players who could bring a fresh eye to the game.

Finally, the close relationship between the Naughty Dog team, our amazing Sony production team, and our Q/A departments was critical to the smooth running of the project in the final stages of development, as the programmers and artists worked around the clock to polish the content and hunt down the last bugs. Good organization, clear and open communication, and lots of all-hands-on-deck, can-do attitude helped us drive smoothly through the switchbacks of Format Q/A; we completed the game at the start of September, hitting our scheduled gold date perfectly.

THE CHECKERED FLAG

We hope that this overview of JAK X: COMBAT RACING's development will prove as useful to you as looking back over it has been to us, in helping develop best practices that let us all maximize the potential of our games—and the teams that make them—by working swiftly and efficiently. We do what we do at the Dog because we love games and the gamers who play them. We are very lucky to have such an amazing team, who worked with great dedication to make JAK X a success—thanks, guys! We can't wait to see what's coming next, as we move into the 2006 consoles and beyond.

4 Markets

3 Days

2 Exhibit Halls

1 Roof

Witness the "Quadruple Play" as Home Phone, Internet, TV and Mobile Phone come together at CTIA WIRELESS 2006.

CTIA WIRELESS 2006®

A Department of **CTIA**–The Wireless Association

The Most Important Technology Event of the Year!

April 5-7, 2006

Las Vegas Convention Center Las Vegas, NV USA

LOOK FOR SCHEDULES CO-LOCATED CONFERENCES EDUCATION HOUSING @ **WWW.CTIA.ORG**

THE BUZZ ON HARMONIX'S GUITAR HERO

HARMONIX MUSIC SYSTEMS WAS FOUNDED BY ALEX RIGOPULOS AND Eran Egozy in 1995. Alex and Eran met at the M.I.T. Media Lab, and their original goal was to use technology to allow non-musicians to experience the joy of playing music. This goal continues to drive all of our development efforts. Collectively, we believe that making music is one of the most exciting and satisfying experiences possible, but one that is denied to most people, given the time and commitment necessary to achieve proficiency on a musical instrument. We were convinced that if we could remove the technical impediments to musical performance, we could create a compelling experience for our audience and build a business around it. It took us a while to realize that video games would be the most effective platform to achieve this vision, but since we made that connection (triggered by our first look at PARAPPA THE RAPPER), we haven't looked back.

Given this goal, our strategy has been to focus on music titles, beginning with FREQUENCY and AMPLITUDE on the PlayStation 2, and continuing with the KARAOKE REVOLUTION series on multiple platforms. (There was also a brief detour into EYETOY: ANTIGRAV, but that's another story.) We learned valuable lessons from the development of each of these titles, and when GUITAR HERO showed up, we were well prepared to tackle it. It wasn't apparent to us until it was done and in the hands of

ILLUSTRATION BY CLIFF SCORSO, BASED ON PHOTO © ALYSSA SCHEINSON

GAME DATA

PUBLISHER
RedOctane

DEVELOPER
Harmonix Music Systems

RELEASE DATE
November 2005

NUMBER OF DEVELOPERS (AT PEAK)
49

LENGTH OF DEVELOPMENT
Nine months

PLATFORM
PlayStation 2

PROJECT SIZE
303,000 lines of code
32,100 lines of script

GUITARS BROKEN DURING DEVELOPMENT
97

GUITARS SET ON FIRE DURING DEVELOPMENT
1

DANIEL SUSSMAN *is a producer at Harmonix Music Systems and has played guitar for 17 years. His primary guitars are a 1990 Gibson Les Paul and a 1967 Gretsch Astro-Jet.*

GREG LOPICCOLO *is the project director for* GUITAR HERO, *and joined Harmonix in 1998. His current bass is a 1974 Fender Precision with Seymour Duncan pickups. He still hasn't beaten* Cowboys From Hell *on expert, and is therefore obliged to hang his head in shame. Send comments about this article to* ***editors@gdmag.com****.*

the game-playing public, but GUITAR HERO is probably the most successful realization of the Harmonix vision to date. We've gotten tons of feedback from musicians and non-musicians alike about how much the game feels like actually playing rock guitar. In its most successful moments, GUITAR HERO crosses the line between gameplay and actual performance. This is a source of great pride to us, given our early conviction that we could achieve this goal and the amount of time and effort that it has taken to realize it in practice.

FROM OUT OF NOWHERE

For Harmonix, the opportunity to create GUITAR HERO was pretty unexpected. RedOctane approached us at exactly the moment we had a team available, and as soon as we put serious thought into it, we realized it was the game we had always wanted to make. We were given a modest budget and a short (nine months) development cycle to work with, but we were equipped with a mature codebase, a lot of relevant design experience, and a huge reservoir of passion and enthusiasm for the subject matter.

Project leader Greg LoPiccolo and audio lead Eric Brosius had played together in Tribe (bassist and guitarist, respectively), a prominent Boston rock band from the distant past. Art director Ryan Lesser had toured the U.S. playing guitar in The Laurels; game systems programmer Dan Schmidt is front man and rhythm guitarist for indie pop group Honest Bob and the Factory-to-Dealer Incentives; and producer Daniel Sussman is currently guitarist for the Acro-Brats, a Boston-based punk band. Many other team members previously had been or are currently in bands. Rock music is a big part of our lives and of the Harmonix company culture, and GUITAR HERO provided a perfect opportunity for us to celebrate and pay tribute to (not to mention poke fun at) the music we love.

WHAT WENT RIGHT

1 **STRONG CONSENSUS ON THEME.** When we started, it was important that the entire team had a unified vision of what this game would be about. Initially, RedOctane just inquired about the possibility of a guitar game. About 10 minutes into our first brainstorming session, we realized that it needed to be a *rock* guitar game. There was unanimous support for a no-holds-barred rock experience, something that could seep into every element of the game, from the music selection, to the art direction, to the HUD design. When confronted with any art or design question, it was great to be able to ask, "Does this rock?" and proceed accordingly. The entire team was committed to this vision, which gave us tremendous focus and saved a lot of time and debate. We also owe a big shout-out to RedOctane, who got the vision immediately and gave us plenty of space to realize it. They were involved in the big design calls, but they demonstrated a lot of faith in our team and let us make the game we wanted to make.

2 **SOLID PRE-PRODUCTION MILESTONE.** We kicked off the first GUITAR HERO milestone in style, gathering all the leads together for an introduction to rock. Greg's brother has a nice pad with a ceiling-mount projection TV and an enormous stereo. It was there that we met up to drink some beer and watch music videos. We spent about three hours watching live Led Zeppelin, AC/DC, Rolling Stones, and lots of other classic material. It sounds cheesy, but getting together and talking about how great Jimmy Page was or how much we loved or hated The Who really had a cohesive effect on the team. Finding out that we were all opinionated music snobs was awesome. From there, we went to work drawing up character and venue concepts and building a playable prototype. Having such a productive pre-production period was a big part of why GUITAR HERO was successful. It's amazing how much of the initial concept work made it into the final product.

These sub-par third-party guitars were casualties of war.

3 **STRONG PROTOTYPE.** We had the multitrack audio files for Weezer's "Dope Nose" from AMPLITUDE and, because it has a guitar solo, we thought it would serve as a decent prototype. Game systems programmer Dan Schmidt wired up a simple 2D display with white lines scrolling down the screen. Then he added a basic scoring system, and all of a sudden everyone on the team was putting high scores up on the white board. Eric Brosius (audio lead) spent some time in his home studio working up some other songs (he did sound-alikes of "Walk This Way," "Back in Black," and "Ain't Talkin' 'Bout Love") and we were hooked. The fact that GUITAR HERO was a compelling play experience very early on in development gave us a lot of confidence that we could make a successful title.

4 **STRONG CODEBASE.** One reason we got GUITAR HERO up and running so quickly was that we were sitting on five years worth of well-architected and well-maintained music game code. We didn't actually reuse any of the beatmatching code from prior titles, but lessons learned from those earlier efforts took a lot of the guesswork out of developing the gameplay core. The venue and character systems we developed for KARAOKE REVOLUTION provided a basis for the GUITAR HERO venue and character systems. There's no way we could have completed GUITAR HERO in nine months without such an advanced and flexible game engine and codebase. Not only did it provide an excellent springboard

for development, but it meant we could be flexible in development. We were able to prototype different game mechanics in days instead of weeks. We also had the luxury of drawing from the experience behind the code. We had already made several 3D rhythm-action games and were aware of some of the potential pitfalls.

5 **GOOD GUITARS.** We knew going in that the title would stand or fall on the strength of the guitar peripheral, which made us extremely nervous, since we knew nothing whatsoever about peripheral development. To their great credit, RedOctane was able to design and manufacture a guitar peripheral that surpassed our expectations. Very early in the process, they asked us for our guitar feature wishlist. We asked for the tilt sensor and whammy bar well before we had any clear idea of how they would be used in the game. They responded that these features would significantly raise the manufacturing cost for the peripheral, to which we replied that we needed them, because they would *kick ass*, even though we couldn't explain exactly how just yet. And they agreed to keep them. Thanks guys!

WHAT WENT WRONG

1 **NO GUITARS.** The development of the guitar controller started at the same time as the game. As a result, a lot of the game was developed on third-party guitar controllers that we could only find over the internet (and in short supply, at that). These controllers were pretty low-grade and we went through a ton of them; there's still a pile of dead plastic guitars in our storage space. The third-party guitars had flaky buttons, no whammy bar, and a strum bar that only worked in one direction, which meant that there was a whole set of features we couldn't test. And they were useless for difficulty tuning. We didn't get our first controller prototype from RedOctane until a few days before E3. Even after that, they came in such small batches that we didn't have enough guitars for every developer and were constantly running up and down the halls borrowing guitars from each other. We also felt pretty strongly that Q/A should be testing with the guitars as much as possible, which put a further strain on the guitar supply. Not having a healthy supply of guitar controllers made it tough for us to thoroughly test both the software and the hardware.

2 **FREESTYLE MODE.** We were really excited about including a freestyle mode so players could assemble their own crazy solos with divebombs, feedback, finger-tapping, and all the other adolescent guitar showboating moves that we so dearly love. We poured a lot of precious development time and resources into this feature, and sadly, had to cut it. It was very ambitious and we simply didn't have the time we needed to both make it sound good and integrate it into gameplay. Some of us feel that it was a gamble worth taking. Others aren't so sure.

3 **SCHEDULING OVERSIGHTS.** For the most part, we were successful in creating a complete and detailed schedule early in the project and sticking to it. However, there were a number of seemingly small and mundane features (such as the unlock store, the intro cut-scene, and the win sequence) that were either underspecified or didn't make it into the schedule at all, and they added up to quite a bit of work. We got blindsided by this at the onset of beta, at which point we were still dropping lots of new content into the game. If the team hadn't been so good, we would have been in serious trouble. This is a classic developer misstep, and one that we've made before. We thought we had learned our lessons and applied the necessary structure to avoid this problem, so it really stung when it cropped up again. As a result of this experience, Harmonix has now instituted a much more rigorous and detailed set of practices for producers and assistant producers, which apply to all of our projects.

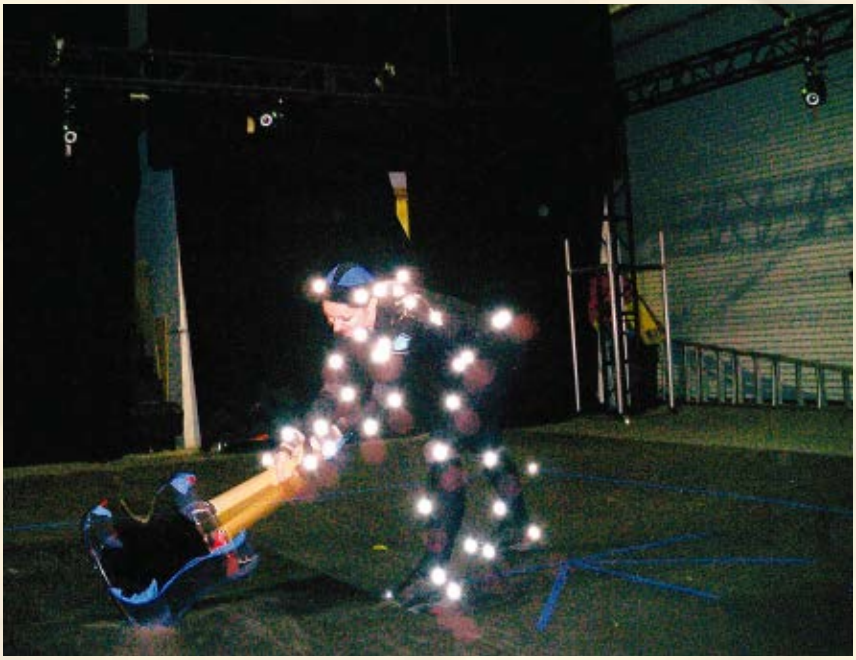
We've since developed a detailed project-scheduling template for use in all Harmonix projects, with a checklist of generic features due at each milestone. Obviously, there's a great deal of content and features which are unique to each title, but we're now much less likely to overlook the mundane features that are common to all of our titles, and we have a much clearer set of guidelines for when in the project timeline such features should be complete (examples include memory card save/load system and icon, having win sequences designed and implemented, and so forth).

None of this is rocket science; they're all features that are not that hard to implement, but if you overlook more than a few, you can get burned pretty badly if your team is on a tight schedule. None of us like working crunch hours, and although we haven't been able to completely avoid them at Harmonix, we are

This would be a good opportunity to use the guitar's whammy bar.

constantly striving to keep them to a minimum. Crunch hours caused by poor planning are a mark of shame which we work hard to avoid. If we had planned for those features better and more completely, we could have implemented them more efficiently and possibly had more space for some of the features that were cut. Which leads us to:

4 **WE HAD TO CUT STUFF.** Good stuff. Stuff like a practice mode that would let you select individual song sections and slow them down so that you could learn them. We did a lot of focus testing and learned that most players were able to pick the game up without much instruction. Based on this experience, we concluded that practice mode was somewhat expendable, though it's a feature that a lot of players have been crying out for.



Harmonix worked with House of Moves for motion capture.

The need for feature cuts was partly dictated by our development philosophy. We had a very ambitious feature set at the beginning, with the knowledge that we might have to trim it as development progressed. We designed what we thought would be a full, deep game and then cut what we needed to in order to finish on time. We think we made the right calls, but some of those cuts really hurt.

5 **NO AC/DC SONGS.** Booooo! Although we were very happy with our final song list, we couldn't get all the songs and bands we wanted. We had a respectable song licensing budget, but there were acts that were beyond our financial reach or simply weren't interested in participating. No need to list them; we all know who they are. Maybe in a sequel? We can always hope.

GOT THE TIME

GUITAR HERO was a resounding win for us, and it wouldn't have come together if we hadn't been properly set up to make it in the time we did. Game projects at Harmonix usually have short development schedules and lots of design mystery, which can be a challenging set of issues to manage effectively. We have evolved some tactics over the last few years to survive in this environment, and they served us well on GUITAR HERO. A few key tactics worth mentioning:

Low barriers to communication between all disciplines and all levels of management, coupled with aggressive attempts to facilitate communication. Anyone can talk to anyone else about anything, in newsgroups, in meetings, or over lunch. It's understood that everyone on the team is responsible for the total experience, no matter what their role is. Lots of suggestions and complaints are made, most get rejected, but the good stuff usually ends up in the game.

Large, professional in-house Q/A team. We generally don't have time to rely overly on publisher Q/A. We need quick turnaround on new builds, and instant communication between the developers and Q/A staff to identify and address issues as fast as possible. We usually assume from the onset of any project that we are going to do the bulk of Q/A in-house. We try to empower our Q/A teams to participate actively in the design process and to be the conscience of their projects. At this point, we couldn't imagine making games without having the Q/A teams integrated this way.

Early prototyping, with lots of review and iteration cycles. At the onset of development, completed design docs are not always very useful to us, as we're not yet sure what will be fun. A perfect case in point for GUITAR HERO was the integration of Star Power and the whammy bar. We tried a number of different modulation effects and gameplay roles before settling on the shipped design. In retrospect, it seems like an obvious implementation, but it wasn't obvious to us beforehand. It took several months of constant experimentation and revision before we settled on the final design.

Because of the compressed development cycle, we had to be merciless about keeping our project scope under control. This was made easier by our shared conviction (learned the hard way on previous titles) that if development resources are limited, it is crucial to focus on the core experience and eliminate resource expenditure on anything that's peripheral to the core. Even so, it was somewhat of a white-knuckle ride down the home stretch. We burned time on some features that didn't work out, we cut some features that we wished we didn't have to, but the rock'n'roll experience we were hoping for survived pretty much intact.

KNOCK 'EM DEAD, KID

We on the GUITAR HERO team are all somewhat astonished that we actually got paid to make this game. Scary moments notwithstanding, it was pure fun from beginning to end. Since the launch, it has been gratifying and shocking to discover how many people (including many who are way out of our core demographic) share our juvenile dreams of rock glory. It really doesn't make much sense that a little plastic guitar should make you feel like such a badass, but somehow, it does. We learned some surprising things about fun, passion, and suspension of disbelief while working on this game, and we can't wait to apply those lessons to our next title. *Hint:* It won't feature white belts or guitars above the waist. ::

MICK WEST

THE INNER PRODUCT

MULTI-CORE PROCESSORS

NEXT-GENERATION GAME PLATFORMS are increasingly making use of multiple processors to deliver more computing power. The Xbox 360 has three symmetrical CPUs running two hardware threads each. The PlayStation 3 has one master CPU with seven separate "SPE" processors. Specs have yet to be released for the Nintendo Revolution, but I suspect that it will be dual threaded. Newer PCs built for playing games have dual core hyper-threaded processors.

Programmers who are used to programming for a single core are now faced with the challenge of translating their programming skills to a multi-core system. Ideally, you want to fully utilize the power of all the processors available to you. A processor idling is a processor wasted.

In this article I describe some of the issues involved with programming software for multi-processor machines and discuss a few potential ways that developers might architect their engines to more fully utilize the potential power.

DON'T DO IT?

The simplest option is to just ignore the additional CPU resources and run all your code from a single thread on a single CPU. (See Figure 1.)

Note that the "rendering" portion here is the CPU contribution to rendering (such as high-level visibility determination and preparing the data stream for the GPU to process). The GPU processing is not shown.

Here the flow of execution is very straightforward. We have the game physics, followed by the game logic, and finally the CPU contribution to rendering. Each of these very high-level tasks also has a strict order of mid-level tasks. The physics, for example, first processes the rigid bodies, then the IK animation, and finally the particle systems. This set of systems is a simplification of what you might find in a current single processor game engine.

While the single threaded option is obviously not going to take advantage of any of the additional processing power, it does greatly simplify the task of programming the game. Whether this is the best choice depends very much on the type of game you are programming, your schedule, and the programming resources available to you.

Remember, the GPU is still responsible for a large portion of the visual quality of your game. If the game contains a relatively low amount of moving and interacting objects (such as a golf game), then you may well be able to get away with using a single processor.

The decision to remain single threaded can also greatly simplify the debugging process. Multi-threaded programs can be very difficult to debug. If your schedule is short, and if you already have a single-threaded code base up and running, then it may well be cost effective to simply ignore the extra CPUs, at least for this version of the game.

MANUALLY THREADED

Some mid-level tasks are sufficiently decoupled that they can be executed in any order, and even in parallel. If you are able to identify two such tasks, then by running them in parallel, you will be able to save the processing time taken by the shorter task, since the total execution time will be the time taken by the longer task. In Figure 2 the second core has been partially utilized. The rigid body physics and the IK animation are now running concurrently. The time saved here is again the shorter of the two tasks (the IK animation).

The logic portion of the pipeline is not threaded. Typically, code like AI and game object logic are very dependent on the order of execution, and so it's difficult to split an existing AI system into separate threads without introducing a large number of interesting bugs.

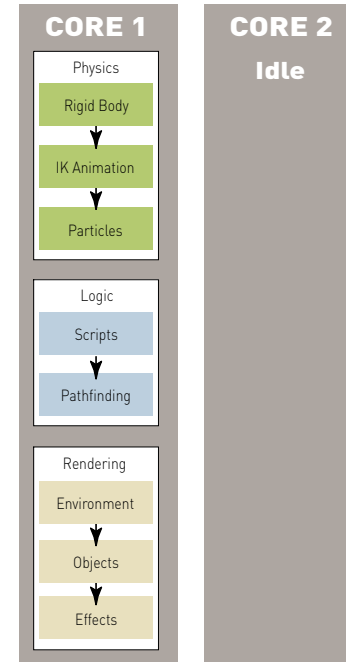


FIGURE 1 **A single thread with one core idle.**

Rendering also has some parallelism (shown in Figure 2), with the object rendering running at the same time as the environment rendering.

This approach of selectively shifting certain large tasks into separate threads (and onto separate CPU cores), is something you might do as a quick and easy step when retrofitting an existing engine to take advantage of multiple cores. The implementation will vary greatly according to the dependencies between systems in your engine. For example, it might be impossible to perform environment rendering at the same time as object rendering if they share some resource, such as a pool of vertex buffers, which is not thread safe.

The disadvantage is that you have a fairly limited number of pieces of processing with which to play. There are very few ways in which you can arrange things so that your engine will still work,

MICK WEST *was a co-founder of Neversoft Entertainment. He's been in the game industry for 17 years and currently works as a technical consultant. Email him at mwest@gdmag.com.*

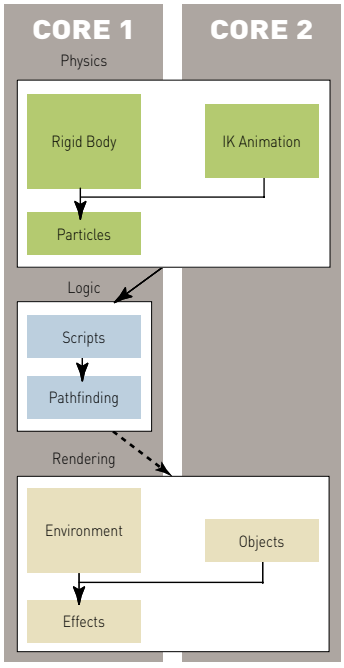


FIGURE 2 **Some systems are manually shifted to the second core.**

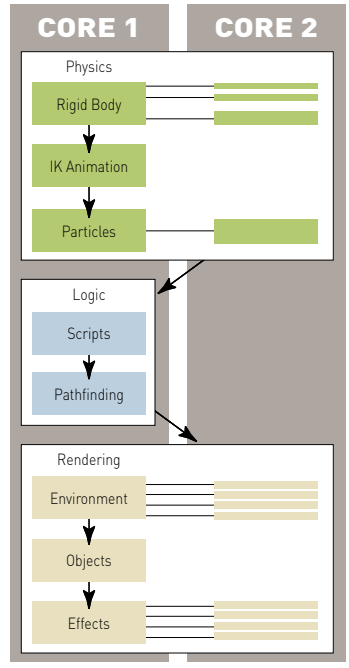


FIGURE 3 **Systems are forked at the object level so the main order of execution is preserved.**

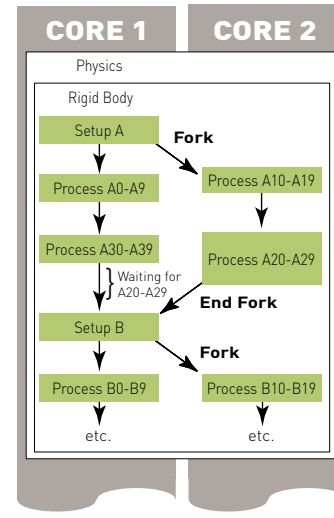


FIGURE 4 **Each forked thread processes a few objects at a time. Objects might be processed out of order.**

and hence you will end up making poor use of the second CPU core.

Since the parallel tasks are doing very different things, you will also not be making the best use of your shared cache on a hyper-threaded system.

SINGLE THREAD FORKING

If your high-level systems are not suitable for running in parallel, then you should turn to the low level. Many systems execute a small set of code over a large number of objects. If the order of processing of the objects is not important, you can start a very short-lived second thread to process half the objects.

You can visualize this as your main thread "forking" into two (or more) threads, each thread processing some of the data, and skipping over data processed by other threads, until all the data is processed. The forked thread(s) then terminate, leaving the main thread to continue along the normal path of execution.

Figure 3 shows how the flow of execution works. The main thread in Core 1 is identical to our simplistic single threaded "solution" in Figure 1. The only difference is that when the main thread is able to process a number of objects concurrently, it forks to process those objects, then continues as a single thread.

Maintaining the original order of execution is a great advantage in simplifying the interactions between systems in your engine. This kind of execution also makes better use of shared caches, since the two threads are running the same code and operate on nearby data.

Figure 4 shows how the threads might be forked down at the object level. To reduce overhead, the forks normally process batches of objects. Here we are processing 10 objects at a time. First we see a brief setup phase for one set of objects of type A. Then both cores simultaneously begin processing objects, with Core 1 processing objects A0 through A9 and Core 2 processing A10 to A19.

In this example, the processing of objects takes different amounts of time. Core 2 finishes its batch first and immediately starts on objects A20 to A29. Shortly after that, Core 1 quickly starts and finishes off A30 through A39.

You can clearly see that the order of processing the objects cannot be guaranteed. It will probably not be the same from one frame to the next and will very likely differ from one supposedly identical run to the next (say, with input recorded for a demo). This can lead to obscure bugs when objects are mostly processed in one order, but very occasionally are processed in a slightly different order, revealing a bug that is very difficult to track down.

If you can work with that, then this technique is very powerful. It's especially nice in that it scales very well to larger numbers of CPU cores. If you want to target the Xbox 360, then you can simply fork six times for the six hardware threads on the three cores of the console.

SIDESTEPPING AMDAHL

All the methods so far suffer from Amdahl's Law. Amdahl noted that a large portion of a program execution is unavoidably serial, so you're limited by the time it takes to execute that code, no matter over how many processors you spread the execution of your parallel code.

It's possible to sidestep Amdahl's Law to some degree by splitting your code into multiple parts and running them in parallel in what is essentially a multi-stage pipeline.

Figure 5 (pg. 32) shows how this procedure looks during a single frame of execution. Both cores are 100 percent utilized, with Core 1 running logic and rendering, while Core 2 is fully devoted to physics.

Obviously you're going to run into problems if you try to run logic and rendering on an object in the middle of its physics update. The solution here is to double buffer the output of the physics, and have the logic and rendering actually work with the results of the physics simulation from the previous frame.

Figure 6 (pg. 32) shows how this works over the course of three frames. It is easier to understand if you start on the

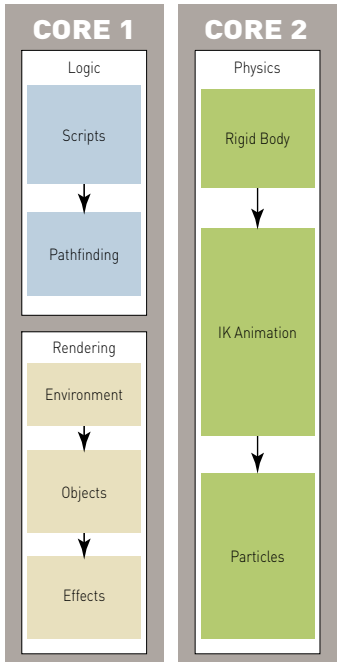


FIGURE 5 **Physics runs concurrently with logic and rendering, utilizing nearly 100 percent of Core 2.**

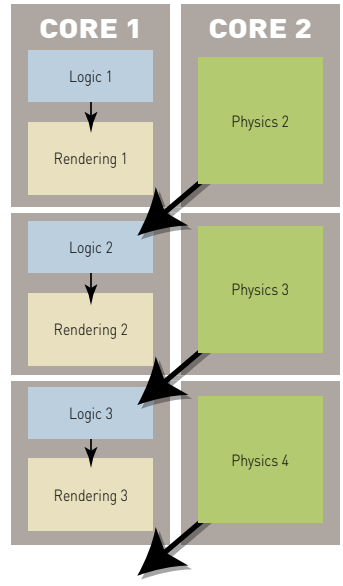


FIGURE 6 **Physics results are double-buffered and used by logic and rendering on the next frame. This sidesteps Amdahl's Law.**

second frame. The logic and rendering is running on Core 1 using the physics data that was calculated in the first frame. At the same time, Core 2 is running the physics simulation that will be used by the logic in frame 3.

This architecture should be very familiar to anyone who knows how rendering works on a console such as the PlayStation 2. There, the GPU is effectively another processor and (usually) runs in parallel with the CPU, rendering the graphics that were prepared by the CPU on the previous frame. Here, we are just extending the pipeline.

In theory you could separate this pipeline even more if you have additional CPU cores to play with. You could pipeline logic and rendering on two separate threads, so rendering for frame 1 happens at the same time as logic for frame 2 and the physics for frame 3.

But there are number of problems with extending it in this way. First, you likely won't find that the various systems take the same amount of time to execute, so you'll be wasting some CPU power. I've artificially shown the physics taking the same time as logic and rendering combined, but the idea here is that physics usage by the game designers will expand to fill the physics processing capability available.

A potentially more serious problem is the introduction of lag into the game. With a multi-stage pipeline, it could take several frames for the users' input to reach the screen. Even on a single threaded game, it will take on average 2.5 times the duration of a frame for the input of the player to affect the image seen on screen.

That's usually acceptable, but if you add another stage to the pipeline, then it jumps to 3.5 times the duration of a frame. At that point if you are running at 30fps (shame on you), then it could take a maximum of 4x1/30 seconds, or 133 milliseconds. That kind of lag is barely acceptable on the internet. Running at 60fps cuts your lag down to 66ms.

PPU AND GPU

A PPU is a physics processing unit, basically a very fast second processor dedicated to the types of computations that are frequently used in physics simulations. A PPU can take the place of Core 2 in Figures 5 and 6. A PPU (currently only available as an add-on card for PCs) should be able to handle vastly more objects than a general purpose CPU core.

A GPU is a graphics processing unit. But modern GPUs have become so incredibly powerful that it's possible to do some physics processing on them, too. The problem with GPUs is they are designed to run as a one-way pipe at the end of the pipeline. It's not really efficient to take the result of calculations from the GPU and feed them back into the start of the pipe.

However, what you can do is separate out the part of the physics that is primarily for cosmetic effect and need not affect the rest of world. Particles systems, explosions, and debris, for example, still need physical simulation to move convincingly in the game world. The GPU can take over the task of the minimal movement and collision calculations required by these "physics effects" and incorporate that at the rendering stage. Since this all takes place on the GPU it greatly reduces the bandwidth required to render huge amounts of physics effects.

The best of both worlds would be a graphics card that contains both a PPU and a GPU.

WHAT TO DO?

A hybrid approach may work very well. There are more complex ways of doing this, and there are many other considerations, but combining the physics pipelining of Figure 5 with the task forking of Figure 3 will allow you to tune your approach to various target architectures in a reasonably straightforward manner. Pipelining your physics in a separate thread will also allow your engine to eventually take advantage of a PPU processor.

RESOURCES

Wu, David. "Physics in Parallel: Simulation on 7th Generation Hardware."
www.cmpevents.com/Sessions/GD/PhysicsinParallel.ppt

Intel Corporation. "Threading Methodology: Principles and Practices."
http://tinyurl.com/atsb5

Gabb, Henry and Adam Lake. "Threaded 3D Game Engine Basics," Gamasutra.com.
www.gamasutra.com/features/20051117/gabb_01.shtml

STEVE THEODORE

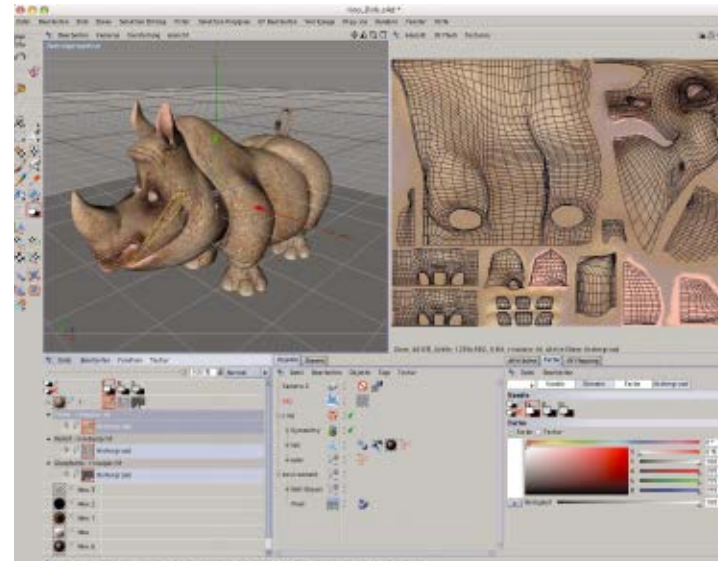
DEMOLICIOUS DERBY

A cornucopia of downloads you really ought to check out

AT THE RISK OF DATING MYSELF, I'LL candidly admit that my first 3D software package required me to build scenes in text files, and I can still remember when all the modeling and animation packages on the market could be counted on the fingers of one hand.

Then, in the pioneering days of CG, every Siggraph and NAB show was as enticing as a fully-stocked candy store. Keeping up with the progress was tricky, since in the early 1990s software developers had little hesitation in charging $50,000 for a fully loaded animation package. Everyone in the business, though, followed the various packages avidly, lusting after proprietary features, obsessing over interface differences, and marching off with bands and banners to every new platform war.

Fifteen years down the road, the market is immeasurably bigger but a lot less passionate. While there must be a few students and games industry wannabes who have time to dissect every press release that makes its way to Highend3d.com, most of us in the game industry have let our old hobbyist impulses molder. Grizzled veterans like myself are heavily invested in our primary packages. We're overstressed with production deadlines and we've seen enough over-hyped software that we no longer believe in magical one-button solutions. Let's face it. Professional artist snobbery is a powerful factor. When all the art on a vendor's web site is obviously done by hobbyists (or worse, programmers), it's easy to dismiss a new piece of software.



BodyPaint 3D is a module within Cinema 4D from Maxon Software.

THE 'HOWEVER' SECTION

Of course, with this kind of buildup, astute readers of this column will be expecting a "however," and here it is.

However, our professional lives are inextricably tied to the ongoing evolution of 3D software, and staying current with the tools and their development is a key part of our profession for three reasons.

First, reconnecting with the enthusiasm that kept us up nights experimenting with UV texturing tools while our peers experimented with Jello shots can be a lot of fun, but it's also a good career move. Keeping an eye on new tools needs to be part of your primary skill set. Think of it as professional self-defense.

As a concrete example, consider the meteoric rise of Zbrush, which has gone from being an obscure hobbyist tool to an industry trendsetter in just a few years. The industry as a whole has certainly benefited from such an injection of energy and new techniques. However, more than a few people who had staked their careers on fairly esoteric technical skills (like the ability to NURBS model a convincing head) found that their status as character modeling geniuses got knocked down a few pegs once lightning-fast subdivision modeling became available to the masses.

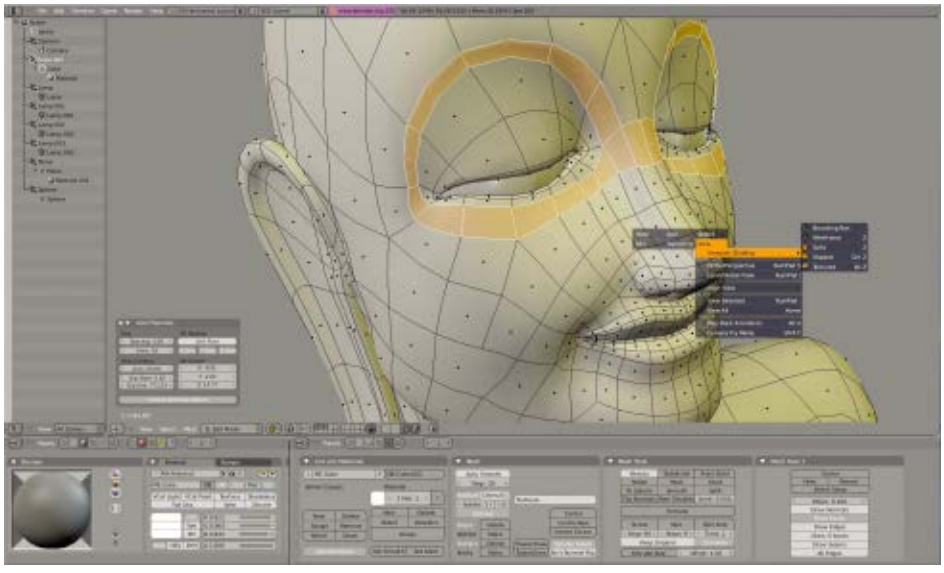
Naturally, the best artists win out in the long run—but the run is a lot longer for those who can't keep up with the times. So if pure intellectual curiosity doesn't drive you to download every new graphics demo that comes along, a little enlightened self-interest might.

The second reason to subject yourself to weeks of playing "Where's my hotkey?" with new tools is that nothing gives you a better perspective of the strengths and weaknesses of your own workflow than being forced into a different way of working for a while. It's all too easy to believe that the complications of doing something the way you've always done it are natural

STEVE THEODORE *started animating on a text-only mainframe renderer and then moved on to work on games such as* HALF-LIFE *and* COUNTER-STRIKE. *He can be reached at* **stheodore@gdmag.com**.

laws of the universe. In fact, the hoops you jump through every day may really be the result of a hastily tacked-on feature, or misunderstood programming spec, or garden-variety incompetence on the part of your package vendor.

Finally, meeting a good implementation of a feature you've only known in a half-assed version is an eye-opening experience. It can also teach you a lot more about what's really going on in the depths of your package than many hours of patient lecturing from your coders (assuming you can find a coder who lectures patiently). And yes, sometimes having access to a piece of software that few other artists know about enhances your reputation for working miracles. While all of us at *Game Developer* are firmly of the opinion that artists who keep their techniques secret are bad for the industry, there's no denying that sometimes having an obscure little ace in the hole can add a little oomph to your reputation.



Blender is an open-source alternative to many 3D art packages.

GOLDEN AGE OF THE DEMO

Luckily for us, this is a great time for the experimentally minded artist. The spread of 3D graphics hardware and its accompanying standards has made building a 3D application a lot simpler than it used to be. Techniques that used to be academically esoteric are now familiar to many coders, so it's far easier to build, say, a subdivision modeler today than ever before. On top of all this, the graphics market has grown enormously, adding game companies, web designers, and pro-sumer hobbyists to the traditional film and broadcast houses. The upshot is that even smallish developers are producing pretty impressive products.

Another big reason we're living in the golden age of art-tool experimentation is the easy availability of free demos. In the days when an animation package cost more than a new BMW, getting your hands on a software seat was a major undertaking. However, since the release of the Maya Personal Learning Edition and Gmax, vendors have come to realize that building a cadre of users is a key way to drive future sales. Add in easy broadband access, and suddenly you can have hands-on experience with almost any important tool in the industry for just 25 cents worth of disk space and a couple of hours watching a progress bar.

With all this in place, I want to point out a few useful demos that you might want to check out. Keep in mind that the information below isn't intended as a review. Reviews are supposed to help your purchasing decisions—the notes below are only intended to signpost packages that might be interesting and informative for you to experiment with. If a package you think is worthwhile doesn't appear on the list, please send me an email and I'll mention it in a future column.

$15,000 WORTH OF SOFTWARE ... FOR FREE!

There's so much going on in the art tools space right now that I couldn't possibly cover every worthwhile demo in a single column. Nevertheless, here is a list of end-to-end packages—integrated modeling/animation/rendering packages that are intended to be the main pump in an art pipeline. Although Max and Maya dominate the game industry, there are a half-dozen other packages that provide similar functionality, equally advanced technology, and (most important for our purposes) free demos. And, if you've spent your career using either Max or Maya without every stepping foot on the other side, it's not a bad idea to download a learning edition demo to see how the other half lives.

So here, without further ado and in alphabetical order, are eight free demos you ought to investigate.

3DS MAX AND GMAX, AUTODESK

www.autodesk.com

Autodesk's web site offers a complete trial download of 3ds Max 8, which is time-limited to 30 days. The demo is identical to a licensed copy for the duration of the trial period. Until recently, Autodesk also offered Gmax, a free but feature-limited version of Max oriented toward modeling and level building. Autodesk stopped offering Gmax in October of last year, but you probably could find a copy out there if you looked—it's very popular among various mod communities. Windows 2000/XP only.

BLENDER, BLENDER FOUNDATION

www.blender.org

Blender doesn't, strictly speaking, offer a demo. It doesn't have to, because it's a free open-source project. Though "open source" is often a synonym with "amateurish," Blender is pretty slick and innovative. It also offers special support for game developers with integrated Python scripting, GL native graphics, and in-window interface overlays. The complete, unlimited version of the latest Blender build (and the complete source code as well) is available for free under the Gnu public license at Blender.org. If you get some value out of Blender, consider supporting the Blender Foundation with a donation, or encourage your company to underwrite the project. As befits an open-source package, it's available for Windows, Mac OS X, Linux, and Solaris.

CINEMA 4D 9.5, MAXON SOFTWARE

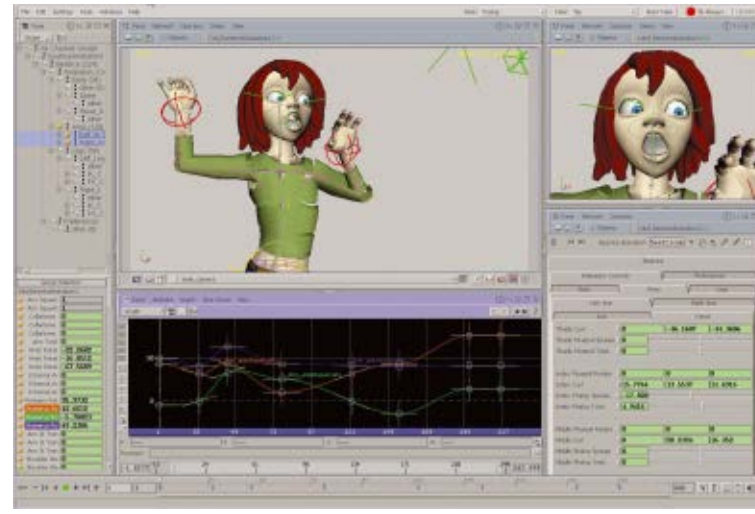
www.maxon.net

Cinema 4D is a modular modeling, animation, and rendering package that is particularly popular in Europe. It sports a highly configurable interface, sophisticated rendering, and an integrated scripting language. One especially interesting module is BodyPaint 2.5, an integrated 3D paint package. A feature-complete demo, Cinema 4D's trial is nearly identical to the shipping product but with saves disabled, and is available for both Windows and Mac OS X from Maxon's web site.

HOUDINI APPRENTICE, SIDEFX SOFTWARE

www.sidefx.com

Houdini, from SideFX Software, has deep roots in the film and television business but hasn't ever been a major player in making games. Houdini combines familiar modeling, animation, and rendering tools with a very strong procedural orientation—almost any modeling, rigging, or animation task can be controlled with a flowchart-like network of parametric controls. Another draw is a robust built-in image compositing system, a key part of the package's appeal in the film and broadcast markets.



Houdini Apprentice allows artists to experiment for free with software from SideFX.

The Houdini Apprentice program can produce watermarked images up to 640x480 and offers a locked file format. Downloads and accompanying documentation are available at the SideFX web site for Windows XP and Linux.

LIGHTWAVE DISCOVERY EDITION, NEWTEK

www.newtek.com

LightWave has long had a reputation for producing innovative tools with decidedly different flavor. While many Max or Maya veterans may find the modular LightWave workflow unfamiliar, it does give some valuable perspective on the evolution of the standard all-in-one graphics package, and a lot of interesting interface features. Newtek offers a free Discovery Edition of the package, which is limited to low vertcount models and watermarks its images. Unfortunately, finding a copy of the Discovery Edition for download is surprisingly hard. Luckily, you can instead shell out less than $50 for one of several introductory books, which includes a demo version on CD.

MAYA PERSONAL LEARNING EDITION, ALIAS

www.alias.com

Maya's Personal Learning Edition is a restricted version of Maya Complete. Images rendered inside the package are watermarked and limited to 1,024x768 resolution. It offers access to all the basic Maya tools, but doesn't allow the use of plug-ins or model data. However, you cannot access the Mel scripting language. The Personal Learning Edition will read Maya complete files, but only writes encrypted .mp format files that can't be read by other versions of Maya. The Personal Learning Edition is available for Windows 2000/XP and Mac OS X.

SOFTIMAGE XSI MOD TOOL, SOFTIMAGE

www.softimage.com

In the days of the $40,000 workstation, Softimage was one of the dominant mainstream software packages, offering, among other things, one of the very earliest commercial implementations of inverse kinematics. With the release of Softimage XSI in 2000, the Montreal-based developer has been pushing hard to reclaim a spot in the limelight. Now in version 5.0, XSI features a very robust toolset and a very modern, if somewhat unusual, front end.

Softimage's demo version is known as the Mod Tool. It produces watermarked images (limited to 510x510 pixels) and, like the Maya Personal Learning Edition, it uses a special locked model format for saving work and has a variety of safeguards to prevent it being used as a substitute for a full licensed seat. The Mod Tool also supports export plug-ins that will write out models for Epic's Unreal Engine and Valve's Source Engine directly. As of this writing, the Mod Tool still reflects the feature set of SoftImage version 4.2 (the current shipping version is 5.0). The Mod Tool is available for download at the Softimage web site for Windows 2000/XP.

JESSE HARLIN

AURAL FIXATION

Q/A FOR AUDIO

Dynamic themes were an important part of STAR WARS EPISODE III: REVENGE OF THE SITH's universe.

A GREAT GAME SCORE IS ONLY AS GREAT as its implementation. Unfortunately, composers are only closely involved in music implementation a small percentage of the time. The task of wiring a game's soundtrack frequently falls to audio leads, music supervisors, music implementers, programmers, or any combination thereof. In-house composers find they have a greater chance of tackling implementation, while freelance composers will most likely find themselves only peripherally involved.

Additionally, music implementers are faced with their own set of problems. For the most part, music is a somewhat nebulous aspect of a game's construction. There is a largely unspoken guideline among Q/A departments that states, "If music is playing, everything's working. If music is not playing, something's broken."

According to Michael Ward, Q/A tester for LucasArts, "No other aspect of video game development requires more open communication between testers and developers than audio. If the line of communication is not established, then there is a blind spot between the developer's intentions and the tester's intuition." Frequently, this line of communication is never substantially established and so the burden of all Q/A work for music rests solely on the shoulders of the implementer/composer.

The result is often that the composer hands control of the music to a team of people that may or may not include any musicians, who are entirely responsible for the bug-free implementation of the soundtrack. With such a scenario occurring so frequently, it's little wonder why repetition fatigue, playback bugs, and unmusical execution are still common issues in game music development.

WITH A LITTLE HELP FROM MY FRIENDS

There are, however, two very specific steps that can be taken to help composers and implementers get more ears listening for bugs and free up communication regarding the intended music implementation.

Forget for a moment the "If music is playing" axiom of test. The truth is that Q/A is the last line of defense for all bugs between the developers and the consumer. Testers spend every day neck-deep in the game ravenously looking for problems. With a little bit of guidance, testers can become a composer/implementer's best friend. They simply need to be told what to listen for.

The best way to do this is to enlist the aid of a dedicated audio tester. Most test teams, whether at a large publisher or a small developer, have someone with some kind of musical background or an ear for sound. The goal is to first identify this person and then request that, at the very least, he or she set aside a few hours with each new build of the game to go over audio issues, and only audio issues. This centralizes the point of contact for audio bugs, allowing the developers to highly educate one person on the intended implementation. The second step is to give the tester the information he or she needs in order to find bugs.

X MARKS THE SPOT

In 2004, LucasArts developed STAR WARS REPUBLIC COMMANDO. To give the game a cinematic feel, we developed a complex interactive music system that reacted to AI activity, scripted events, and had branching gameplay. As such, the music implementation was often closely tied to issues that were completely out of my control. Small changes to props, doors, or enemy counts would sometimes result in bugs like music loops that never ended or broken transitions between cues.

To help fight the battle against music bugs, I enlisted the help of two audio testers. I spent a few days writing up what I named a "music map." The music map was in essence a blueprint, a Microsoft Word document that specifically detailed every instance of music throughout REPUBLIC COMMANDO and where each music change occurred. For example, one specific instruction read, "If battle music is playing, walking into room D will cause a victory flourish to play, followed by a new suspense cue."

With everything detailed, Q/A simply had to read through the music map and write up bugs when the music failed to respond as it was documented.

A few months later, the level designers at The Collective were handling the music implementation for STAR WARS: EPISODE III REVENGE OF THE SITH. Again, I spent a few days producing a detailed music map of all of the intended music changes and then handed it off to both The Collective and our dedicated audio testers. This time, not only did the remote music implementers have detailed instructions of where music was to be wired, but Q/A also had a way of checking to ensure that everything was wired correctly—all of which was centrally located in one document. When music wasn't properly wired, bugs were written up and assigned to the level designer in question, creating an official request for music to be fixed.

Although control over the music assets was out of my hands, both level designers and Q/A were following my documented intentions as specific guidelines. The result was a small team of people all working together toward perfect music implementation.

TAKE THE TIME

Dedicated audio testers and music maps work toward removing the barriers that commonly block communication about music implementation. By taking the time to enlist others in the fight against bugs and to create a centralized music map, composers/implementers need not endeavor to implement a great game score alone.

JESSE HARLIN *has been composing music for games since 1999. He is currently the staff composer for LucasArts. You can email him at jharlin@gdmag.com.*

Join the Pulse of Gaming™

JOBS.MIDWAY.COM

Midway Games Inc. is a leading developer and publisher of interactive entertainment software for all major video game systems, releasing some of the most popular games in video game history.

Our goal is to create the most compelling, high-quality games around. We're always searching for world-class talent with energy, commitment to excellence and unrestricted imaginations to join our exceptional teams.

FIND OUT HOW YOU CAN BECOME A PART OF THE MIDWAY PULSE!

OPENINGS AT THE FOLLOWING LOCATIONS:

- ***SEATTLE***
- ***LOS ANGELES***
- ***SAN DIEGO***
- ***AUSTIN***
- ***CHICAGO***
- ***NEWCASTLE***

POSITIONS

Character Artists
Sr. & Lead Concept Artists
Sr. & Lead Environment Artists
FX & Compositing Artists
Sr. & Lead Animators
Lighting Technical Director
Technical Artists
Project Creative Director
Sr. & Lead Game Designers
Gameplay Programmers
Network Programmers
A.I. Programmers
Sr. Online Programmers
Sr. Software Engineers
Sr. & Lead Tools Programmers
Technical Directors
Advanced Technology Producers
Associate Producers
Audio Director
Localization Manager
Q.A. Analysts
Q.A. Supervisors

COME VISIT US IN BOOTH 1824 AT THE GDC.

LOG ON TODAY!

MIDWAY

NOAH FALSTEIN

GAME SHUI

WHEN THE HUNTER BECOMES THE HUNTED

I GREW UP WATCHING OLD WORLD WAR II movies. I loved the submarine warfare scenes in which the captain would stealthily sneak up on the enemy convoy, lining up a shot on an oil tanker or ammunition transport. "Fire one! Fire two! Torpedoes away!"

Then invariably: "Enemy Destroyer closing, crash dive!" And the klaxon would sound. Then the tense wait for the depth charges would begin.

THE RULE

Make the hunter become the hunted.

Encourage the player to shift roles abruptly between predator and prey. This can build dramatic tension, encourage daring strategies, and even serve as comic relief when threatening enemies become vulnerable. But make sure the circumstances for the switch feel fair to the player and not arbitrary or vindictive.

THE DOMAIN

This rule's domain is all games involving conflict or battle.

THE RULE TRUMPS

I can't find rules that directly trump this one, but several reinforce or support it. "Raise the emotional stakes to maximize the player's involvement in the game" (see "A Matter of Life and Death," December 2004) is a more general case of the same principle, and so is "Fight player fatigue." That you need to make the rule feel fair to the player suggests another rule for a future column.

EXAMPLES AND COUNTEREXAMPLES

This is a rule that has been used in a huge number of games. One of the first and clearest instances is good old PAC-MAN. When the power-up dots instantly change the marauding and dangerous ghosts into vulnerable (and valuable) targets, the hunters become the hunted. More recently, KATAMARI DAMACY evokes laughs as once ominous enemies scurry away, shrieking in panic as your katamari grows in size.

Real-time and turn-based strategy games have long depended on the rock/paper/scissors principle in which one specific military unit may be devastating to another but be vulnerable to a third. That's precisely what made those submarine movies so exciting. Often, it's not just the units, but the terrain or mode those units are in that adds to the excitement.

A WWII submarine 300 feet down was relatively safe but was unable to threaten others. Only by coming close to the surface could it become dangerous—and vulnerable. STARCRAFT made use of mode change with units that could cloak or hide until they attack, just as the *Star Trek* movies and games use cloaking devices.

The THIEF series—and in fact the whole sub-genre of stealth games—also uses this rule upfront. An assassin striking under the cover of darkness is the medieval world's equivalent of an attack submarine. Many RPGs use magic to the same effect. A spellcaster is often very powerful under the right conditions, but let the ninja with a sword get close and he's just magician sushi.

IMPLEMENTATION

The best implementations of this rule encourage players to maximize their own risk, rather than have that risk thrust upon them. A submarine commander or stealthy assassin can choose to stay hidden if the prey seems too heavily guarded; but if the reward is great enough, the risk of attack may be justified. Then if the attack fails, the player blames his or her own greed and not the game design.

NOAH FALSTEIN *is a 25-year veteran of the game industry. His web site, www.theinspiracy.com, has a description of The 400 Project, the basis for these columns. Also at that site is a list of the game design rules collected so far and tips on how to use them. Email him at* ***nfalstein@gdmag.com****.*

FEEDBACK ON FEEDBACK

"NEGATIVE FEEDBACK" (OCTOBER 2005) produced a flurry of email. Neil Meskauskis of Sonicube Games pointed out some other examples, like the fact that in BATTLEFIELD 2, the larger the caliber of the gun, the heavier it is and the slower you run. But he warned too that inconsistent negative feedback can leave the player "frustrated and feeling cheated."

Several readers mentioned sports games, a particular blind spot for me, so I'm grateful for the examples. Charles Naut pointed out that some sports games use dynamic difficulty adjustment. The opposing team plays better against you if you are winning and eases up if you are behind.

Paul Terry of EA who had worked on many of the HIGH HEAT BASEBALL games for 3DO mentioned negative feedback is often present in sports games, citing how they model pitcher fatigue. Players can't overuse their star pitchers without consequences. The game designers applied real-world constraints, having pitchers become fatigued not only within a game, but over a season if they were played too often.

Intriguingly, Terry also mentioned that EA fixed another problem with pitching by more accurately modeling real life. That in turn applies to this column, as the use of rock/paper/scissors relationships in the real world can provide great hunter/hunted relationships that feel completely fair because they are familiar to the player.


ACTIVISION.
activision.com
THIS IS YOUR GAME
• Programmers
• Game Designers
• Scripters
• Producers
• Artists
• Art Directors
• Animators
• Sound Engineers
• Marketing
• Creative Services
• Sales
• IT
Please visit us at the GDC Career Pavilion at Booth 1809
ACTIVISION. VALUE PUBLISHING, INC.
NEVERSOFT
infinity ward
Vicarious Visions
RAVEN
Luxoflux
BEENOX
ShABA
TREYARCH
ZAXIS
Toys For BOB
FOR A COMPLETE LISTING OF OPENINGS AND TO APPLY,
VISIT WWW.ACTIVISION.COM (GO TO "CAREERS").

CLARINDA MERRIPEN

'Overhead' is dead.
—Tom Peters

BUSINESS LEVEL

MAXIMIZING OPERATIONS

What business geeks can do for your game studio

BUSINESS PEOPLE HAVE A HAND IN everything that makes your studio function, from managing your money (finance), to managing your employees (human resources). They project your professional image (marketing), they engage your customers (business development), and they enable your product (IT).

Whether you know it or not, we business geeks make or break your game studio. If you look at why most studios have gone out of business in the past few years—really scratch below the surface—you'll find poor operations at the center of the problem. Bad operations kill a studio, but great ones add value.

WHAT GOES AROUND, COMES AROUND

What is spent on operations will come back to you. The following examples are all scenarios in which operational costs could be turned into profit centers.

In the first example, a group of internal marketers threw themselves between the publisher's marketing plan and the studio's. These marketers helped balance art needs between the publisher and the game developers. When an unexpected request for an E3 trailer arose, they negotiated a separate art contract and turned it into additional profit for the studio.

A second example involves an IT person who did more than just support and maintain servers, finding a way to streamline build times by 40 percent. By exploring innovative and cost-effective server technology, this IT superstar wound up saving each developer roughly 30 minutes of compile time.

Third, consider what an HR generalist could do if he or she built up relationships with the local college. I knew one who negotiated a wholly-owned research and development grant that the studio used to reap the technological work of a professor and his grad students for a year.

NO SECOND STRINGERS

In game development studios, we require amazing work from our line production people. We would never think of hiring someone off the street to manage our art department, or someone who only programmed in basic to code an engine in C++. But when many game studios think about staffing their operations department, they put any old Dick or Jane in control.

How often have you seen programmers running IT in their spare time, or an administrative assistant running HR and, god forbid, all the finances? Likewise, business development often falls to the overworked, ill-equipped CEO. And marketing, if it's thought about at all, is typically pushed to the design department and artists. Someone, usually reluctantly, designs a web site once, and occasionally throws something newsworthy on it. Anything beyond that is considered wasted time.

Understand that "who" matters. Just as you have rock star artists, designers, and programmers, you need to have operational rock stars. Demand no less.

PLANNING VS. EXECUTION

Start-up companies typically can't afford to hire one person for each operational function. But operations are first about planning, then executing. From the birth of your studio, you need to know what you want to do in each area—not an organic "we'll cross that bridge when we come to it" idea, but an honest-to-goodness strategic vision of what you are going to do.

In finance, the very first thing you need to understand is the difference between bookkeeping and managing a studio's money. From day one, you need to think about who will be filling the role of CFO and who will create the projections for different financial scenarios every time they arise. That's finance. Bookkeeping, on the other hand, is part of what your accountant should do.

Second, you need to decide who will be your financial consultant when it comes to your profits (where will they come from?) and alternative funding sources (how will you finance projects or make bi-weekly payroll?). Don't take it for granted that just anyone can do it. You need someone who loves money and numbers and understands real markets versus paper ones.

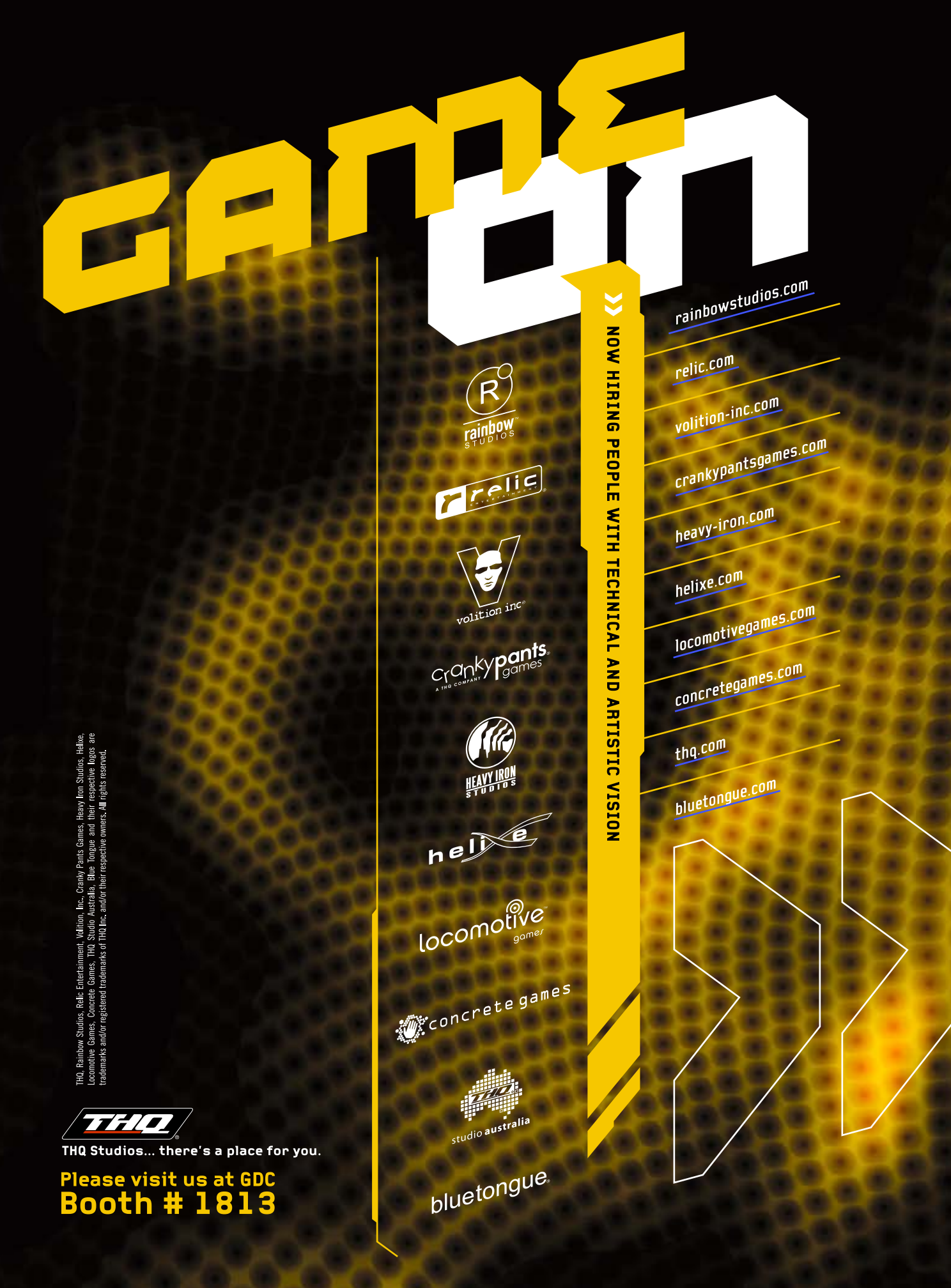
If you can't hire someone right away, there are alternatives. Go to a business school, find some brilliant grad students, and pay them to help out. Ask amazing financiers to be on your board of directors—and listen to them. Use consultants; tap into one of the boatloads of temporary CFO/controllers out in the big wide world.

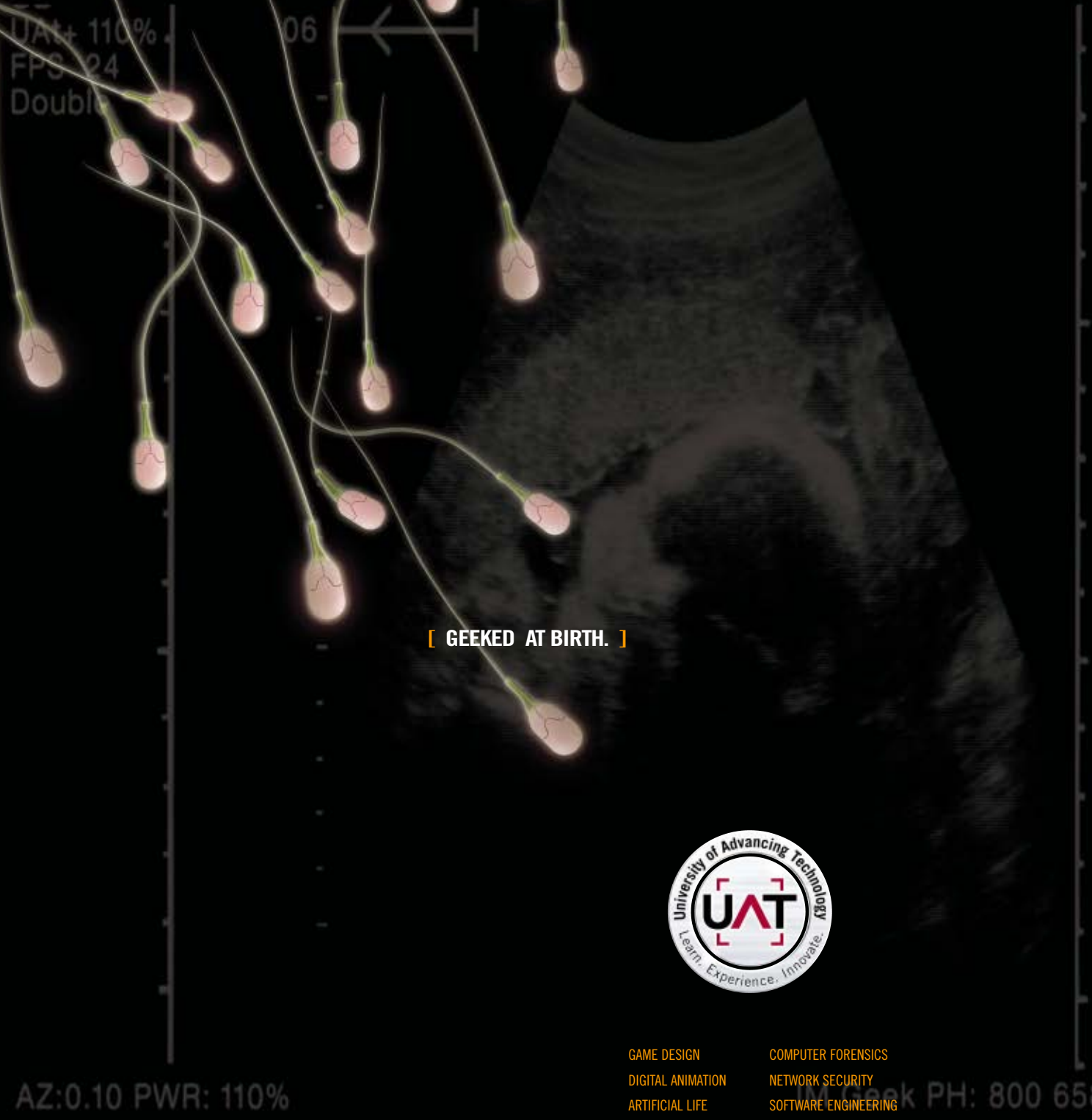
BE THE BEST OR BE CUT

The ultimate test for the people you have in operations is very harsh: Would another company pay for their services? If they are not the best and brightest, and don't add the most value, outsource their jobs and be done with it. If their recruiting doesn't bring in the absolute best, hire contract recruiters. If your business development fails to bring enough contracts, work with agents.

Operations personnel must be smart, fiscally responsible, and adventurous. And as game development studio owners, you simply shouldn't hire anyone who's below that bar. The policy of hiring only the most talented or promising people to make your games should extend equally into the rest of your business. Remember, what you put in is what you'll get out.

CLARINDA MERRIPEN *is vice president of operations at Cryptic Studios. She's moderating with real live business geeks at this year's GDC. Touch base with her at cmerripen@gdmag.com.*


GAME ON
NOW HIRING PEOPLE WITH TECHNICAL AND ARTISTIC VISION
rainbowstudios.com
relic.com
volition-inc.com
crankypantsgames.com
heavy-iron.com
helixe.com
locomotivegames.com
concretegames.com
thq.com
bluetongue.com
rainbow STUDIOS
relic ENTERTAINMENT
volition inc
crankypants games
A THQ COMPANY
HEAVY IRON STUDIOS
helixe
locomotive games
concrete games
studio australia
bluetongue
THQ, Rainbow Studios, Relic Entertainment, Volition, Inc., Cranky Pants Games, Heavy Iron Studios, Helixe, Locomotive Games, Concrete Games, THQ Studio Australia, Blue Tongue and their respective logos are trademarks and/or registered trademarks of THQ Inc. and/or their respective owners. All rights reserved.
THQ
THQ Studios... there's a place for you.
Please visit us at GDC
Booth # 1813



[GEEKED AT BIRTH.]

University of Advancing Technology
UAT
Learn. Experience. Innovate.

GAME DESIGN
DIGITAL ANIMATION
ARTIFICIAL LIFE
DIGITAL VIDEO
WEB DESIGN
COMPUTER FORENSICS
NETWORK SECURITY
SOFTWARE ENGINEERING
TECHNOLOGY MANAGEMENT
GAME PROGRAMMING

You can talk the talk. Can you walk the walk? Here's a chance to prove it. Please geek responsibly. **www.uat.edu >** 800.658.5744

At the **DigiPen Institute of Technology**, we believe that there are no shortcuts to a serious career in the field of digital interactive entertainment. Developing the knowledge and ability to achieve the professional standards demanded by employers is an extremely difficult path.

The DigiPen Institute of Technology brings together all of the critical elements necessary to give students the best opportunity for success. For more information, visit **www.digipen.edu** or come speak to representatives from DigiPen at **GDC in San Jose, CA (Booth 1142).**

DigiPen congratulates IGF "Innovation in Game Design" finalist *Rumble Box* (DigiPen student project 2005).

5001 150TH AVE NE REDMOND, WA 98052 • PHONE: (425) 558-0299 • FAX: (425) 558-0378 • WWW.DIGIPEN.EDU

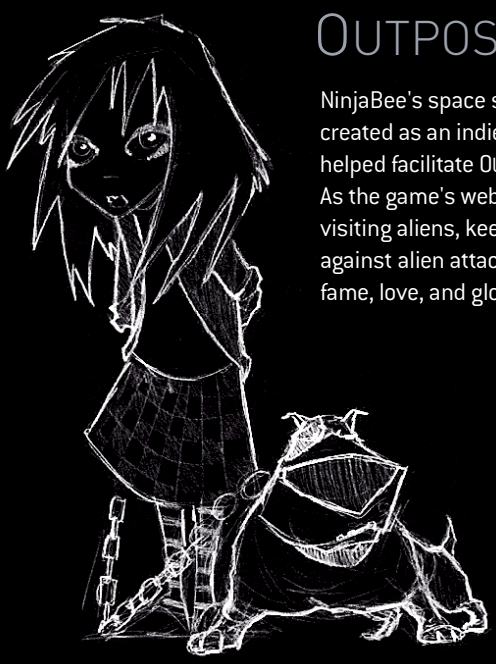
ADVERTISER INDEX

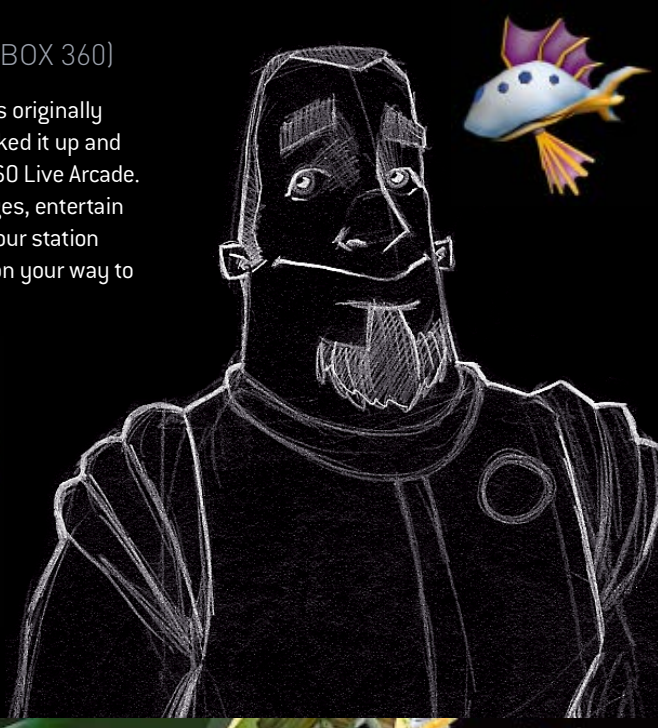
Game Developer (ISSN 1073-922X) is published monthly by CMP Media LLC, 600 Harrison St., 6th Fl., San Francisco, CA 94107, (415) 947-6000. Please direct advertising and editorial inquiries to this address. Canadian Registered for GST as CMP Media LLC, GST No. R13288078, Customer No. 2116057, Agreement No. 40011901. **Subscription Rates:** Subscription rate for the U.S. is $49.95 for twelve issues. Countries outside the U.S. must be prepaid in U.S. funds drawn on a U.S. bank or via credit card. Canada/Mexico: $69.95; all other countries: $99.95 (issues shipped via air delivery). Periodical postage paid at San Francisco, CA and additional mailing offices. **Postmaster:** Send address changes to Game Developer, P.O. Box 1274, Skokie, IL 60076-8274. **Customer Service:** For subscription orders and changes of address, call toll-free in the U.S. (800) 250-2429 or fax (847) 647-5972. All other countries call (1) (847) 647-5928 or fax (1) (847) 647-5972. Send payments to *Game Developer*, P.O. Box 1274, Skokie, IL 60076-8274. For back issues write to *Game Developer*, 1601 W. 23rd St. Ste. 200, Lawrence, KS 66046-2703. Call toll-free in the U.S./Canada (800) 444-4881 or fax (785) 841-2624. All other countries call (1) (785) 841-1631 or fax (1) (785) 841-2624. Please remember to indicate *Game Developer* on any correspondence.

A THOUSAND WORDS

OUTPOST KALOKI X (XBOX 360)

NinjaBee's space station sim OUTPOST KALOKI was originally created as an indie PC title, before Microsoft picked it up and helped facilitate OUTPOST KALOKI X for the Xbox 360 Live Arcade. As the game's web site explains: "Solve challenges, entertain visiting aliens, keep your ratings high, defend your station against alien attackers, and rake in the money on your way to fame, love, and glory."





ART DIRECTOR: BRENT FOX
CONCEPT/2D ART: SHAWN BOYLES
3D MODELER: JEFF WIGGINS

"To call the graphics **amazing** is an extraordinary understatement." GameSpy

"Oblivion is, at this time, the **best-looking game** I have ever seen in my life." Xbox.com

"One look at Oblivion will **shatter your conceptions** about what is possible in a video game." GameInformer

their secret weapon? Gamebryo

To create their most stunning game so far, Bethesda Softworks used a mix of genius, hard work — and the most flexible 3D graphics engine on the market. **The Elder Scrolls® IV: Oblivion,™** a title for Xbox 360 and the PC, is already being hailed as one of the most beautiful games ever. How can Gamebryo help your vision emerge? Find out at **emergentgametech.com**.

Gamebryo

EMERGENT GAME TECHNOLOGIES

Miles Solid Gold Hits

RAD GAME TOOLS ©

THE BEST IN GAME DEVELOPMENT TECHNOLOGY

BINK VIDEO — *Bink Video Technology*

GRANNY 3D — *Granny 3D Animation System*

Pixomatic Software Renderer