



CMP
United Business Media

» PRODUCT REVIEWS PHOTOSHOP CS3 EXTENDED » SPACENAVIGATOR

AUGUST 2007

gamedeveloper

THE LEADING GAME INDUSTRY MAGAZINE

» **CASE FOR THE TECH ARTIST**

HOW AND WHY STUDIOS
NEED THESE OXYMORONS

» **CODERS MOUTH OFF**

EXCLUSIVE ROUNDTABLE
WITH FIVE PROGRAMMERS

» **THE INNER PRODUCT**

PAINTING WITH DOMAIN
SPECIFIC LANGUAGES

POSTMORTEM:
FINAL FANTASY XII



CREATE

In *Assassin's Creed*, Ubisoft used Autodesk® 3ds Max® software to create a hero character so real you can almost feel the coarseness of his tunic.

ANIMATE

Autodesk® MotionBuilder™ software enabled the assassin to fluidly jump from rooftops to cobblestone streets with ease.

INTEGRATE

Using Autodesk® HumanIK® middleware, Ubisoft grounded the assassin in his 12th century boots and his run-time environment.

**HOW UBISOFT GAVE
AN ASSASSIN HIS SOUL.**

autodesk.com/Games

Image courtesy of Ubisoft

Autodesk, MotionBuilder, HumanIK and 3ds Max are registered trademarks of Autodesk, Inc., in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. © 2007 Autodesk, Inc. All rights reserved.





22

POSTMORTEM

22 FINAL FANTASY XII

The twelfth installment in Square Enix's wildly popular series was developed in a fashion quite unlike the others. No one on the team had worked on a major FINAL FANTASY title before, and most had been accustomed to mid-sized projects, not massive ones. General manager of R&D Taku Murata focuses more on what went wrong in this ingenious postmortem.

By Taku Murata

DEPARTMENTS

2 GAME PLAN *By Brandon Sheffield*

An E3 Without Developers

4 HEADS UP DISPLAY

Game developers' census, the big three at E3, and more.

28 SKUNK WORKS *By Tom Carroll*

3DConnexion's SpaceNavigator, Adobe Photoshop CS3 Extended, and product news.

FEATURES

7 PROGRAMMERS' ROUNDTABLE

In this exclusive roundtable interview, five top technophiles discuss what's right and wrong with game programming these days, from in-house procedures to multi-processor systems to code reuse.

With Brad Bulkley, Stuart Denman, Chris Hecker, Clinton Keith, and Bruce Rogers

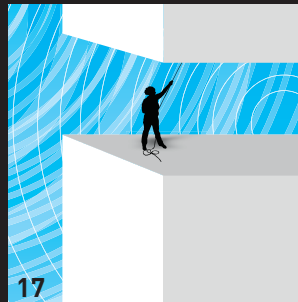
17 THE CODE/ART DIVIDE: HOW TECHNICAL ARTISTS BRIDGE THE GAP

Game studio Volition (THE PUNISHER, SAINTS ROW) makes a case for hiring a few technical artists, their value ranging from time savings to smoother in-house communication. As linchpins, technical artists keep programmers and artists bonded in their mutual cause, helping to ensure that each department gets what it needs from the other—and within reason according to the project's schedule.

By Jason Hayes



7



17

COLUMNS

33 THE INNER PRODUCT *By Mick West*

Domain-Specific Languages

[PROGRAMMING]

39 PIXEL PUSHER *By Steve Theodore*

Fight the Good Fight

[ART]

43 GAME SHUI *By Noah Falstein*

Hybrid Vigor

[DESIGN]

44 AURAL FIXATION *By Jesse Harlin*

Vox Populi

[SOUND]

64 BUSINESS LEVEL *By Nicholas Belliaeff*

More of What They Want

[BUSINESS]

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. All content, copyright *Game Developer* magazine/CMP Media, LLC, unless otherwise indicated. Don't steal any of it.



AN E3 WITHOUT DEVELOPERS

THIS YEAR'S E3 WAS ONE OF THE ODDER GAME

events I've been to over the years, due in large part to the fact that very few of you were there.

I spent most of my time either trudging through the streets of Santa Monica, determined not to cancel or postpone any appointments (sorry, id Software), or sitting in my hotel, clacking away on a company-loaned Powerbook, trying to turn my high-level interviews into low-level sensationalist news stories for the game blogs to Hoover up.

It was a decidedly different E3. Gone were the glitz and glamour of previous years—no booth babes, no loud music, no swag. And yet, for all its bells and whistles, in those days I found myself typing on the floor of the press room most of the time, rather than from the balcony of a swanky Best Western. (Any hotel is a luxury to me.) I found the new experience alternately more classy and less classy, depending on the moment.

WHERE HAVE THE TEDDIES GONE?

Many of these changes were welcome, to be sure, but at the same time, I felt a marked lack of developer presence. In the past, it wouldn't be uncommon to see a group of scrappy Q/A-types wandering around in a pack, crashing rival game demos, or designers and senior producers demonstrating their games to round men with teddy bear backpacks. Now, only a very small percentage of the low-thousands of people in attendance were in-the-trenches developers.

Not only did this make my job harder in terms of reporting on the event for Gamasutra (we get shafted enough as it is for not being a consumer publication), it also made me reconsider the aims of the new E3 as a whole.

The ESA represents publishers, and that was clearer than ever this year. I know that publishers are necessary for a large portion of the industry, but the focus at E3 was even less on developers. It was on the end products, and I'm not sure it's in the industry's best interest to separate the two.

The fact that the main event was spread out over a two-mile stretch didn't help. There could have been plenty of developers there that I just wasn't able to find, but that would be another problem.

Then there was Barker Hangar, theoretically the games showcase area. Given that it was five miles away from the main drag, it was sparsely attended, even though the indie and serious games showcases were there. It felt like the back of the bus.

HAND-HOLDING

Some folks, like the guys I met from id (after rescheduling—sorry again!), found the show to be a resounding success, with meetings aplenty and no foot traffic to bog them down. Others, like EA, were less convinced, wondering where the retail, investor, and business development meetings had all gone.

From my perspective, I came away knowing just as little about the upcoming crop of games as I did going in. My hands-on time with games was extremely limited, and not just by time constraints. A lot of games were simply on display and not meant for actual play, and the nature of the event just didn't lend itself to casually checking out new stuff. Meetings were booked specifically well before the show, and in some cases, you had to be escorted through a given company's show space by a PR representative.

Any developer who managed to get to the show in the first place, and then leave his or her hotel, would have had a hard time seeing what the competition was up to.

Penny Arcade, which generously let us publish one of its comics this month (see Game Shui, page 43), mentioned that E3 seemed to be just as well "attended" by watching the live video streamed from the event. From a gamer's perspective, the observation was spot on.

THE CIRCUS COMES TO TOWN

New publisher Gamecock had a carnivalesque funeral march for E3 after the final day, with booth babes made up like corpses, masked actors on stilts, black umbrellas, mourning widows, and a eulogy from a man in a gray suit, suspenders, and a blonde wig.

I didn't really feel like E3 was going to disappear, but it was certainly not going to be like this again next year. Too many important people were disappointed. For my part, I managed to make it work, and met some interesting people. But in this case, it was a matter of poking holes in the drab exterior of the show, rather than soaking in energy from a bombardment of games, people, and experiences that the old E3 represented.

I can't decide what was better for me, but I'm sure it wasn't better for you. After all, you developers weren't there, were you? ❖

Brandon Sheffield
Senior Editor

CMP Media, 600 Harrison St., 6th FL.,
San Francisco, CA 94107
t: 415.947.6000 f: 415.947.6090

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

EDITORIAL
PUBLISHER
Simon Carless scarless@gdmag.com
MANAGING EDITOR
Jill Duffy jduffy@gdmag.com
SENIOR EDITOR
Brandon Sheffield bshellfield@gdmag.com
ART DIRECTOR
Cliff Scorso cscorso@gdmag.com
CONTRIBUTING EDITORS
Jesse Harlin jharlin@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 Microsoft
Brad Bulkley Neversoft
Clinton Keith High Moon Studios
Ryan Lesser Harmonix
Mark DeLoura Ubisoft

ADVERTISING SALES
DIRECTOR OF SALES
Steve McGill smcgill@cmp.com t: 415.947.6217
GLOBAL SALES MANAGER, RECRUITMENT & EDUCATION
Aaron Murawski amurawski@cmp.com t: 415.947.6227
SR. EVENTS ACCOUNT MANAGER, SOUTHWEST
Jasmin Davé jdave@cmp.com t: 415.947.6226
SR. ACCOUNT MANAGER, EAST COAST, EUROPE & EASTERN CANADA
Cecily Herbst cherbst@cmp.com t: 415.947.6215
ACCOUNT MANAGER, WESTERN CANADA, INDIA, AUSTRALIA, & ASIA
Amanda Mae Miller ammiller@cmp.com t: 785.838.7523
SALES REPRESENTATIVE, EDUCATION & RECRUITMENT
Gina Gross ggross@cmp.com t: 415.947.6241
MEDIA ACCOUNT MANAGER
John Watson jmwatson@cmp.com t: 415.947.2224

ADVERTISING PRODUCTION
ADVERTISING PRODUCTION MANAGER Kevin Chancel
REPRINTS
PARS INTERNATIONAL
Joe Nunziata t:212.221.9595 e: reprints@parsintl.com
MARKETING
MARKETING MANAGER
Hilary McVicker hmcvicker@cmp.com t: 415.947.6207

CMP GAME GROUP
SENIOR VP, TECHNOLOGY INNOVATORS GROUP Philip Chapnick
VP, STRATEGIC MARKETING Michele Maguire
GROUP DIRECTOR Kathy Schoback

AUDIENCE DEVELOPMENT
GROUP DIRECTOR Carolyn Giroux cgiroux@cmp.com
DIRECTOR Mary Griffin mkggriffin@cmp.com
ASSISTANT MANAGER John Slesinski jslesinski@cmp.com
LIST RENTAL Merit Direct LLC t: 914.368.1000

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

CMP TECHNOLOGY MANAGEMENT
PRESIDENT AND CEO Steve Weitzner
EXECUTIVE VP AND CFO Adam Marder
PRESIDENT, CHANNEL GROUP Robert Faletra
PRESIDENT, CMP TECHNOLOGY INNOVATORS GROUP
Paul Miller
PRESIDENT, CMP BUSINESS TECHNOLOGY GROUP Tony Uphoff
CORPORATE SENIOR VP SALES Anne Marie Miller
SENIOR VP, CMP INTEGRATED MARKETING SOLUTIONS Joseph Braue
SENIOR VP, HUMAN RESOURCES Marvlieu Jolla Hall
SENIOR VP, MANUFACTURING Marie Myers
SENIOR VP, COMMUNICATIONS Alexandra Raine
VP, INTERNATIONAL BUSINESS DEVELOPMENT Patrick Brennan



Game Developer
is BPA approved



CMP
United Business Media



TESTING SUPERSTAR!



Seapine

TestTrack[®] TCM

The ultimate tool for test case planning, execution, and tracking.

How can you ship with confidence if you don't have the tools in place to document, repeat, and quantify your testing effort? The fact is, you can't. **TestTrack TCM can help.**

In TestTrack TCM you have the tool you need to write and manage thousands of test cases, select sets of tests to run against builds, and process the pass/fail results using your development workflow.

With TestTrack TCM driving your QA process, you'll know what has been tested, what hasn't, and how much effort remains to ship a quality product. Deliver with confidence—be a testing superstar!

- Ensure all steps are executed, and in the same order, for more consistent testing.
- Know instantly which test cases have been executed, what your coverage is, and how much testing remains.
- Track test case execution times to calculate how much time is required to test your applications.
- Streamline the QA > Fix > Re-test cycle by pushing test failures immediately into the defect management workflow.
- Cost-effectively implement an auditable quality assurance program.

Seapine ALM Solutions:

TestTrack Pro 
Issue & Defect Management

TestTrack TCM 
Test Case Planning & Tracking

Surround SCM 
Configuration Management

QA Wizard Pro 
Automated Functional Testing

Download your **FREE** fully functional evaluation software now
at www.seapine.com/gametcm or call 1-888-683-6456.

 Seapine Software™

THE BIG THREE AT E3

E3, now officially dubbed the E3 Media & Business Summit, may have been downsized on the whole, but the big three press conferences from Nintendo, Sony, and Microsoft were true to the old formula. —Brandon Sheffield



MICROSOFT

Microsoft's conference, held in a high school auditorium on the evening before the first day of the proper show, focused primarily on upcoming games and existing services, rather than large new announcements. The presentation was led by then-Xbox corporate vice president Peter Moore (now head of EA Sports), who boasted the 360's to-date sales of 5.6 million units, as well as the fact that third-party titles for the 360 had entered top-10 U.S. sales 18 times since the launch of the console versus zero times for the competitors.

Microsoft announced a home console version of the DVD trivia game *Scene It*, which will ship with a new, more accessible remote-like controller and will be supported by more games in the future, according to company statements. In terms of new announcements, Microsoft revealed that a green Xbox 360 will become available in tandem with the much-hyped launch of *HALO 3*; the company also said it has signed a deal with Disney to release high-definition movie content on the Xbox Live Marketplace video download service.

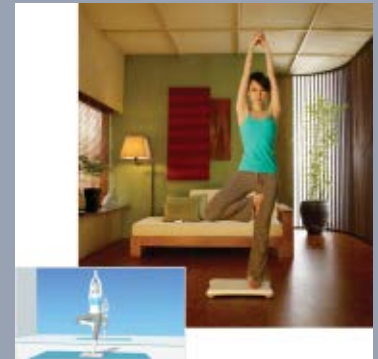
SONY

Sony, for its part, showed an impressive lineup of games and announcements, though the company currently represents the trailing console on all formats. To that point, Sony discussed its already-leaked \$100 price drop for the PlayStation 3's 60GB version, as well as a lighter and cheaper PSP with streaming TV-out for video. The system will be released in packs (of various distinctions) that include a game, a 1GB memory stick, and UMD movie for \$200 beginning this September.

Sony also discussed its PlayStation Network service, announcing new downloadable games such as *WIPEOUT HD* and *ECHOCHROME*, which Sony executives say are unique in that they utilize the system's hard drive, giving them a leg up on the competition. Phil Harrison, president of Sony's worldwide studios also presented *HOME*, the company's virtual world. The service will double as a social network, with content accessible from mobile phones and a web site, as well as the PlayStation Network.

NINTENDO

Nintendo's press conference aimed squarely at the resounding success of its Wii and DS platforms so far in the global marketplace, stating that 69 percent of the game industry's growth over the last year stems from those devices. The company also presented new footage of its biggest upcoming game, *SUPER MARIO SUNSHINE*, as well as an online multi-player version of *MARIO KART* for Wii.



Nintendo's *Wii FIT* will use a new Balance Board peripheral.

The company also unveiled its *Wii Zapper* peripheral, a plastic housing for the *Wii Remote* and *Nunchaku* controllers for use with shooting games such as Sega's *GHOST SQUAD* and Capcom's *RESIDENT EVIL: UMBRELLA CHRONICLES*.

The biggest announcement for Nintendo was its upcoming game *Wii FIT*, and corresponding controller, the Balance Board. *Wii FIT* is a light exercise and body awareness game that uses the Balance Board to gauge a player's center of gravity and body mass, as well as facilitate exercises such as push-ups and step aerobics. This game is clearly aimed at the broader, less game-oriented market. However, Shigeru Miyamoto, a principal designer of the game and peripheral, was quick to point out that the board has plenty of game potential, citing *Wii FIT*'s embedded skiing game as one example.

CALENDAR

Siggraph

San Diego Convention Center
San Diego
August 5-9
Price: \$75-\$1,125
www.siggraph.org/s2007

GC Developers Conference

Leipzig fairgrounds, Leipziger Messe
Leipzig, Germany
August 20-22
Price: 195-415 euros
www.gcdc.eu

Penny Arcade Expo

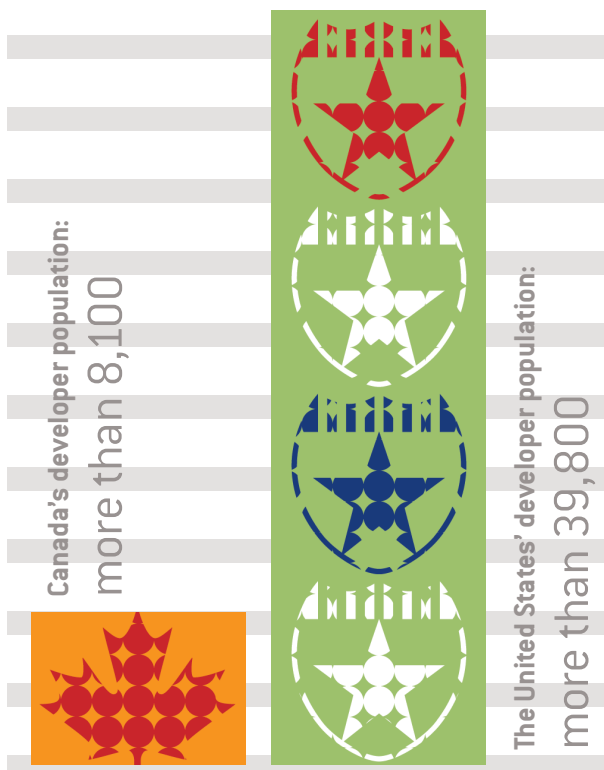
Washington State Convention & Trade Center
Seattle
August 24-26
Price: \$25-\$50
www.pennyarcadeexpo.com

GDC China

Shanghai International Convention Center
Shanghai, China
August 27-29
Price: \$260-\$660
(limited Fair Pass-only entry \$6)
www.gdcchina.cn/en

THE 'BEAN' COUNTERS

Developers number nearly 48,000 across North America



LIKE GUESSING THE NUMBER OF jellybeans in a jar, the number of game developers working in North America had been a mere guesstimate. This summer, though, Game Developer Research, a division of the CMP Game Group (which also owns *Game Developer* magazine), released the first-ever game industry census.

The report, titled "The Game Developer Census 2007," found nearly 39,800 developers in the U.S. plus an addition 8,193 in Canada, putting the total close to 48,000.

The states and provinces that are home to the most game developers are California (more than 46 percent of the U.S. total), Washington (about 12 percent), and Texas (7 percent); and

British Columbia (more than 50 percent of Canada's total), Quebec (26 percent) and Ontario (nearly 16 percent).

The census also suggests that the state of New York has been attracting more and more working developers, cultivating a healthy talent pool there. In 2004, according to unpublished statistics from the *Game Developer* annual salary survey, the state was only ranked fifth (tied with Massachusetts) in terms of number of developers, but now places fourth with about 4.7 percent of the U.S. developer population.

The full report, which includes a state-by-state guide to game industry companies, is available for purchase from www.gamedevresearch.com.

—Jill Duffy

IN THE NEXT ISSUES OF *GAME DEVELOPER* ...

The Inner Product's Mick West tackles responsiveness in programming, September 2007

"Top 20 Publishers," the annual countdown of the top game publishing companies, October 2007

"2007 Front Line Awards," a showcase of the top tools used in game development, January 2008

THE URGE TO CONVERGE

THE DREAM OF MARRYING HOLLYWOOD AND THE game industry is finally coming to fruition, for better or for worse. Steven Spielberg's well-publicized collaboration with EALA for original games has yielded two announcements thus far. Curiously, one game is for the Wii, code-named PQRS, a puzzle game based on the manipulation of blocks, which certainly bucked many expectations of what an EA-Spielberg double-punch ought to be.

The second game, codenamed LMNO, is more of what one might expect from this arrangement—a big-budget action adventure game for Xbox 360, PlayStation 3, and PC that takes the player on an "emotional journey" with a mysterious female, with whom the player's relationship dictates the future. That's the company line, but digging deeper you find that the protagonist is an ex-secret agent and the

way the player treats his or her AI-controlled female companion mainly affects the NPC's special abilities (and thus also the ending). It starts to sound a bit more like a standard game in that light, so the proof will truly be in the pudding. The element Spielberg will be adding is the reason for emotion—he has to make us care about this character.

Just two weeks before this, I heard Clive Barker speak about his own game, JERICHO, an action-based horror game, true to Barker form with unsettling imagery and fantastical story. He's doing what he does best there, and it seems a natural fit, though in this case, there's even more scrutiny being paid to the developer's realization of his vision, since his last game, CLIVE BARKER'S UNDYING, only made it about halfway to the finish line in terms of quality.

Is the convergence of Hollywood and games a

good thing? If I were a designer, I might be somewhat slighted by the intense attention being paid to these apparent filmmaking outsiders. Or are all creative types equal in this industry? I think it's in everyone's best interest to make sure that's the case.

But the fact is, consumers and executives alike are going to pay attention to anyone with an *E.T.* or a *Hellraiser* under their belt. What we as an industry need to do is make sure that in 10 years time, the people who made HALO or SPÖRE are just as well known as the teams turning out these new movie-backed games. And game studios will need to loop these creative minds in where it makes sense, but keep them separate from the project when it comes to certain elements that would be better handled by those who already understand interactivity.

—Brandon Sheffield

gamedeveloper
FRONTLINE
FINALIST '05

"ReplayDIRECTOR is one of the best tools I've found for our test department here at Crystal Dynamics."

■ **Chris Bruno**, QA Manager, Eidos Interactive



BUGS ON ICE

RECORD. REPLAY. FIXED.

ReplayDIRECTOR™ gives you **Deep Recording**. This is much **more than just video capture**. Replay records every line of code that you execute and guarantees that it will Replay with the same path of execution through your code. Every time. Instantly Replay any bug you can find. Seriously.

DEEP RECORDING. NO SOURCE MODS.

download today at www.replaysolutions.com

email us at info@replaysolutions.com

REPLAY SOLUTIONS

1065 E. Hillsdale Blvd. Suite 307 Foster City, CA, 94404 - Tel: 650-472-2208 Fax: 650-472-2210

CRYSTAL
DYNAMICS

eidos

XBOX 360

VIDIA

XBOX



PROGRAMMERS' ROUNDTABLE

▶ **BRAD BULKLEY:** The first topic I want to jump into is cross-platform stuff. One of the big challenges we face at Neversoft is dealing with cross-platform development issues. We share a large portion of our code base across several platforms, and that inevitably leads to technical compromises. But it also forces us to maintain cleaner, more portable code. What's your high-level approach? Do you find it limiting? Do you long for the days of working on one platform?

CLINTON KEITH: I think it's gotten worse. We've gone from a single-processor paradigm to maintaining the difference between the 360 and PS3—it has definitely created a much greater challenge with not only parity, but trying to take advantage of a particular platform over another. The market also makes things a little bit more confused. Before, you could focus on the PS2, which was the weaker machine but the market leader. You could turn your attention to the Xbox after and not worry about it so much.

BULKLEY: Right, whereas now, you have a fight in the marketplace, and machines have power but just in different areas.

BRAD BULKLEY is a lead programmer at Neversoft and a Game Developer advisory board member.

STUART DENMAN is technical director of Surreal Software, a Midway Games studio.

CHRIS HECKER is a technology fellow at Maxis/Electronic Arts currently working on *SPORE*.

CLINTON KEITH is chief technology officer at High Moon Studios and a Game Developer advisory board member.

BRUCE ROGERS is chief technology officer at Cryptic Studios.

Send comments about this article to editors@gdmag.com.

KEITH: We're still in a transition, but the Xbox 360 is the easier and more powerful platform from a traditional programming approach, which may slowly change over time as we adapt to what the Cell architecture can do. But right now, everybody's got a 360 on their desk, and you don't have to bend over backward as much for the 360. That makes the PS3 more of a challenge, especially since there aren't as many out there.

BRUCE ROGERS: I don't know if we're cheating because we don't do PS3 development—we do PC and 360. Tech-wise, they're really similar. The threading issues are the same. You've got multicore on the PC, and you've got three cores on the 360—maybe four and a half if you count the crappy fake ones you get with simultaneous multithreading.

The only issue we have technically between the two systems is that the 360 has a very fixed memory size, which I've actually found to be an advantage when working with artists because [the advantages are apparent to] anyone who has played a PC game and waited 15 minutes while it swapped out its gig and a half of giant textures and wondered why they couldn't just have slightly lower-rezed stuff. It's nice to have that hard limit. In the future when things diverge more—because the PC is constantly getting better and the consoles go in steps—it'll be a little bit more irritating with low-end things. Right now for us [at Cryptic Studios], in terms of the tech of doing cross-platform, it's really not a big deal, but I think that's just because the platforms are so similar. The big issue we have is getting people to test on both platforms because all our tools run on the PC.

KEITH: Same problem here.

BULKLEY: One of the things you guys brought up was multithreading, which obviously brings up a whole host of issues. We have the semi-homogeneous cores of the 360 versus the PPU/SPU system in the PS3. Those two systems don't necessarily lend themselves to the same type of

solution. How do you approach multithreading? Do you try and find some sort of lowest common denominator to have a system that can work with all those platforms, or do you have custom solutions for each one?

STUART DENMAN: For us, it's fairly custom. For the game we're currently working on [at Surreal Software], we're heavily modifying Unreal and ripping out guts and stuff like that. There's kind of a lot of predetermined stuff that comes along with it. We found that we have to port a lot of specialized code off of the GPU and onto SPU code, which is quite a bit different from writing HLSL.

KEITH: We've had the same experience that Stuart's had. It's generally adjusting for the 360, then adjusting a great deal more for the PS3 to work around its limitations. The RSX is a limitation in that it is a bit weaker. We're porting some of that stuff over to the SPUs and then finding some general optimizations for the PS3 that also reflect back to the 360. It certainly hasn't been about finding universal solutions to both platforms.

DENMAN: The 360 has issues too, in how it packs and aligns memory. You kind of have to work around those. But then the PS3 has the two memory pools—the local and system. Typically, we might be heavy in one area of assets that we might put on the Xbox, but on the PS3, we really have to utilize that local RAM as much as possible. Some particular types of assets, like vertices and textures, have to be maximized. That isn't always one-to-one between the systems.

BULKLEY: One of the things we've run into in porting stuff to the SPUs is that we have a lot of programmers who aren't necessarily all working on the same game, and it's difficult to ensure that all their code remains thread-safe. Typically they're not aware of anything else that might be going on outside their system. How do you guys ensure that everyone's code remains thread-safe, and how do you find and fix those issues when they do come up?

ROGERS: I have a training technique I use when we get new people in and they're very gung-ho. They'll say, "Oh, you guys seem really scared of threads, but I did them in college, and they weren't a problem for me!" You know that if someone isn't scared of threads, they haven't done any production code with them.

We threaded our last game on the PC right when the multicore stuff came out, and we basically do everything with thread-safe FIFOs. The idea is that we're pretty much treating it as though it's a separate memory space. There may be occasional read-only shared memory, but if you've got one guy who is modifying a chunk of memory, we never use any semaphores and have two people trying to modify the same chunk at the same time. We set up a FIFO and feed commands to another thread. It's kind of similar to the way the PS3 works, even though we're not doing any PS3 stuff. Basically, we try to avoid it as much as possible, and when we do it, we try to make sure that the communication path between the various threads is very narrow.

BULKLEY: One of the interesting talks I saw at GDC was the guy at Valve who has implemented a lock-free approach. I'm wondering if any of you guys have tried that on consoles. It seems to lend itself more easily to the PC than it would to the PS3, where you don't have homogeneous cores.

ROGERS: Actually, the FIFOs that I was describing were lock-free. It's a pretty easy technique that people have been using for 20 years.

CHRIS HECKER: We [at Maxis/EA] have a lot of problems measuring performance when there are multiple threads. We have a really nice run-time profiler based on Sean Barrett's iprof code that we use that gives dynamic profiling all the time. You don't have to be running VTune and that kind of thing. But because there's no way to do a per-thread RDTSC or anything like that, it's not very effective when there are multiple threads.

You can run without threads; for example, you can turn off the audio thread and that sort of thing, but basically you'll get these wacky profiles coming back from tests, and it's like, "Well, that function doesn't actually do

anything, and it's got this huge blob in there." You'll be wondering, "Is it threads?" It adds this amount of doubt to the system that's hard to compensate for. It allows plausible deniability in some sense by everybody.

“ I've seen the homegrown engines over the life of a company, like 10 years, where legacy code is just a drag on innovation . ”

—Clinton Keith

ROGERS: We've actually been writing a bunch of custom code to deal with that exact problem. We haven't looked into it yet, but Intel has a product that it's selling for the PC—a VTune plug-in—that allows you to do careful thread profiling. If you're doing a PC and console game, usually your threading issues are likely to be similar between platforms, so you can use one to track down the complicated issues on the other.

BULKLEY: Yeah, all the profiling we're able to do on the other cores is because we've written a lot of custom stuff that lets us do real-time profiling of all the functions on all the cores, and we can get histograms so we can see where the spikes were.

HECKER: When you've got SPUs and stuff, it's sometimes easier because they're not swapping between them. You can just say, "Here, I'm going to write some code that specifically pokes into a shared memory location. That's how much this one is using." But on a PC or even an Xbox 360, sometimes the threads are going to switch if you're not setting really hard affinities, especially on the PC, obviously. It's this sea of fog, and you can't actually see what's going on.

DENMAN: Chris, is your timing library sampling, or do you insert begin/end-style timing calls in your functions, and the library handles the context-switches?

HECKER: We insert timing calls, but they don't handle thread switches, which is the problem. Sampling does well if you run it in non-call-attribution mode, but there's a big heavyweight app that you have to start up and it's not installed on all the machines. It's not always running. But we have this thing that's just always running in the background. As long as you're not putting it in the innermost loops, you can just keep it running, and it can pop up this constantly-sorted HUD that's non-intrusive.

From a programming standpoint there's almost no friction to using it. It's always there, and that makes it not just a quantitative change, but a qualitative change in the way you're dealing with performance. In the single-threaded days, you were done. You VTuned occasionally because the profiler does Heisenberg occasionally, but for the most part, you're perfect. Now, threads have set that back five years in terms of knowing what's going on in your code.

ROGERS: One thing you might want to think about when architecting software is the old comment about programmers, which is just because you can do something doesn't mean you should do something. Sometimes you might want to think about a simpler architecture. We're not actually doing this, so I'm basically saying, "Where I'd like to be is better than where I am."



But maybe you just run a command processor and essentially turn all your CPUs into SPUs—basically, run one thread on each CPU, and then have a master CPU that's feeding commands into them. And just ignore the fact that you have shared memory. Then you will have a much easier time tracking all your threading issues because you won't be running multiple threads on each CPU. Things run a little bit faster, but it will of course put a little more overhead on writing your code in the first place.

HECKER: The problem is you have to close the loop with wallclock, right? When you've got a lot of nonlinear threading going on, even if your architecture is set up right, you get things like, "Is there contention happening?" or "Is I/O going slower in this thread because that other thread is blocking it?" Those times when it's nonlinear are when the subtlety and the hard part comes up.

ROGERS: I'm just describing an architecture where you don't ever block on anything. When you're writing the old-school thing where you've only got one CPU or you're writing a one-thread program, you're not expecting to do any blocking. Or if you do, you're blocking in a thread that's doing very little computation. I would suggest an architecture where you have a thread per core, and that thread just runs flat out and processes commands. It puts more complexity into your game application, but it makes it more examinable to tell what's going on. You don't have multiple threads popping in and out at the same time.

KEITH: It's a great opportunity for those who are doing the PS3 engine first. I imagine that if you focus on getting something to run fast on the PS3, then porting it to an architecture like that for other platforms would probably be the best-case scenario.

ROGERS: I kind of wonder if it's in the same way I was describing how the 360 allows you to solve the memory problems on the PC. If you start on the PS3, you might end up with a better engine for the 360 than if you started on the 360, because it forces you to be more rigorous about how you code it.

HECKER: That theory has come up before, but time will tell whether it's actually true.

DENMAN: Taking the existing engine that was made for PC and Xbox 360 and getting it to work on PS3, you have issues. We have particular systems that are written for the SPUs, or written for the CPUs, and likewise on 360, we have systems that are put on particular cores. It's not nearly as nice as being able to have small commands or small jobs that you can just farm off to any thread. It brings the problem of training artists about what it means when one core or system on a particular processor is taking longer. It's a classic CPU versus GPU—what's bottlenecking my frame rate?

We have had to create some visual tools that allow artists to visualize what's going on in different processors, and how if one is taking some milliseconds longer than the other, that's the one you have to target to get your content optimized. The point is that it's different on different platforms, and we have to train our artists to understand that and know that they shouldn't just go hacking models up and reducing all their textures. They need to use these tools to figure out where to attack.

KEITH: It's a big thing, having to educate programmers about that, in terms of what threads are doing what, which are stalling, and where the true bottleneck is coming from, instead of one thread looking like it's taking a lot of time but it's really just stalling on another one.

DENMAN: Exactly. And frame rate is basically an antiquated notion.

HECKER: It's antiquated, except it's the place where the loop is closed finally, right? Like, it's not a very useful tool at all, until the end result when you realize your frame rate is not up. It's the closing of the loop with the player.

DENMAN: Sure, but that's not what the artists need to figure out what to change.

HECKER: True, it's too blunt of a hammer, but it's the one that matters in the very end, and there's a big missing section in the middle where there's a lot of opportunity. Unfortunately, opportunity is made more difficult because there aren't hardware hooks that can even give you the information you need.

BULKLEY: Right. It sort of brings up another problem that I have at Neversoft, which is that at the end of the day, you do have to maintain a certain frame rate, and to do that you have to use CPU budgets early on. One of the big problems I keep running into is how to create those budgets and enforce them throughout the year, when typically, parts of the game won't come online until much later. It's hard to estimate what their impact on the final product will be.

Do you guys have any interesting ways that you either can give accurate estimates for CPU budgets or ways you can enforce them while still being flexible enough to deal with changes to the game?

DENMAN: The enforcement issue is really interesting, especially now with dev kits released on the Xbox being the same as the final kits. You don't have a bunch of memory to experiment with content and throw everything in there. Enforcement's been really tricky. We've set performance and memory budgets, but a lot of time, our estimates don't have a lot of the final information that's needed.

It's like a catch-22—you need the art to measure whether it's going to meet the look of the game, and then once you've measured that, you're like, "Oh, we're over budget. We need to do this or that." So you get in this situation where you constantly have to iterate on your budgets and measure your art. Policing-wise, we have a team of representatives from all the different disciplines that meets every week, which goes over memory and performance and everything. We're constantly trying to police things and keep the bar reasonable.

BULKLEY: Do the content creators—the artists and designers—at your companies work within the final executable? Or do they work like at Neversoft, where we kind of always use a semi-debug executable, which makes it difficult for them to even get an idea of what real-world performance would be?

ROGERS: Actually, here at Cryptic, we ship the debug executable. With an MMO, we always get a lot of crashes out in the field that we can't get crash dumps from. We go through file-by-file, and we have two files: debug, and full

“Almost nobody programs directly to the hardware. It's very frequent now that people are using APIs to talk to very complicated hardware devices.”

—Bruce Rogers



debug. We don't have a release build. The debug build has all the performance-critical files optimized, such that you couldn't get any reasonable call stack off them. But all the game logic is pretty much compiled with full debug. We haven't actually shipped a console game yet, so I'm guessing we'll run into some issues with optimization of quality being a little bit different for a debug build on a PowerPC versus the x86. But it's working okay so far.

BULKLEY: I've heard it suggested that making sure that your development environment is the same as your shipped environment is the way forward. But for us, it's just not quite reasonable.

HECKER: I think that ebbs and flows like a lot of things in the game industry. People go back and forth between separate tools and a separate executable, and then there's the tool built into the executable, and then it comes back into vogue. It goes back and forth.

Doom had a separate thing, and then Underworld had the tool built in and then they swapped. There are pros and cons to both. As you work with one for awhile, you kind of forget the cons of the other and think that you should try it the other way. It's kind of weird because it points to the fact that we're still kind of clueless about how to do this stuff. We're still learning things, but we still keep coming back to it.

There was a great moment at a conference a few years ago. One speaker said, "Yeah, we've got this visual scripting language, but it totally sucks and we're throwing it out and going to a text-based scripting language." And then another person said, "Oh, we have a text-based scripting language and it totally sucks. We're going to a visual scripting language." These were both triple-A, mega-selling games. What's the lesson learned from that?

ROGERS: You have a choice to make, and whichever one you make, it will be the wrong one. But you must choose! What we've learned here, at least by my methodology, is that the most important thing is being able to tell what's going on in the system, like my earlier comment: Can you make the system simpler or use a simpler algorithm instead of a fancy one? If you have crashes, you have to be able to figure out what's causing them. If there was an ideal world, nobody would want to turn off the debugger if it didn't have any effect on performance.

Any single thing that you can do so that you don't have to spend three weeks with your chief graphics programmer trying to track down some tricky bug and just get a call stack emailed to you, saves you so much time. Systems are getting more and more complicated, and yet the problems can still happen at the deepest levels. We don't have perfect abstraction, so we have to know exactly what's broken and why it's broken. Being able to get really good information about what's going on in the system gets more and more critical as time goes by.

KEITH: That's what the embedded tools approach is. We're working pretty much with the shippable game, so when there's a problem introduced, you see it as quickly as possible. The negative side is that it can bring down 100 people with a bad commit. We have a separate thing that we've implemented where build servers that are out there are constantly checking every commit, checking for performance and checking for crashes. You can go out there and

grab any one of the 10 builds that were created over the last day, and the build server will indicate which ones have passed their functional tests.

DENMAN: We have a build server as well, but we also add an element that we call Crash Catcher. It's basically a tool where any time a designer is working on the tools or playing the game, it'll catch crashes and email any dumps.

KEITH: We've got that. Those are hugely beneficial.

DENMAN: Do you guys usually dedicate people to monitoring those as we do?

KEITH: Yes, we have a dedicated team that works on engine and tools, and at least two of the programmers on that team are responsible for addressing those crash logs on a daily basis.

ROGERS: We've basically automated that system, where we have two daily emails that come out, one of which finds any issues that the art and design

teams have made. And every single error that the game can generate knows what file caused that error, and we also know who checked in that file so the email can generate a list of who's to blame. With a program crash, all you have is the call stack, but we have an automated program

that takes the mini-dumps and figures out the check-ins on the call stack. If you're a programmer, you get two emails every day, one with a list of all the data errors, and everybody's name who caused that error.

BULKLEY: So, you're saying to take a look at a function in the call stack and then look those up?

ROGERS: Yes. We have a program we use called Blame. For each line in the code, it'll tell you who checked it in and when. If you have a call stack and there's a check-in from one day ago on a very buggy part of the system, you're pretty sure that's the guy who caused this crash. Sometimes it's more complicated, but it allows us to do really fast triage. As soon as there's an error or a crash anywhere in the system, the system can identify who made it and immediately email that person and his boss.

Then there's a daily email that comes out that basically says, "Here's a bunch of unknown problems that we're having with crashes and asserts and things that have happened," and we track them by how long they take. If they aren't resolved in two weeks, then they get bumped up to another level.

One thing that makes those systems nice is that when the computer's doing it, there's no one to get angry at. It's like everyone becomes a slave to the machine, and says, "I keep getting these emails, so I guess I'll go and fix it."

ROGERS: Is everybody here using a licensed engine?

BULKLEY: No, Neversoft still has an in-house engine. When I started, we used no middleware at all, and we swore we never would. Now, in varying capacities, we use middleware for physics, audio, movie playback, and online. It really became absolutely necessary. We didn't have the people, and we

“ You come back a month later and you realize that you made the same mistakes over again, you're debugging the same exact things that you've fixed before. ”

—Brad Bulkley

THE NEW USER INTERFACE EXPERIENCE



Artist-based **Flash®** UI development

Scalable image and vector graphic animation

High quality rich HTML text output

Efficient Asian font and IME text input

Available on all PC and Console platforms

*"Using our new state of the art CryENGINE 2 and Scaleform GFx,
Crysis will deliver an unrivalled cinematic experience."*

Cevat Yerli, CEO & President of Crytek

*"Mass Effect is only the beginning of a long-term commitment by
BioWare to deliver an incredibly immersive cinematic UI experience."*

Greg Zeschuk, President of BioWare

www.scaleform.com

Scaleform GFx and Scaleform GFC are © 2007 Scaleform Corporation. Flash is a registered trademark of Adobe Systems Incorporated. Crysis and CryENGINE are trademarks of Crytek. All Rights Reserved. Mass Effect is a registered trademark of BioWare Corp. All rights reserved.





didn't have the time to develop and stay current with those technologies. At some point, we just had to make a choice. Were we going to continue to maintain our own engine, or were we going to give up?

ROGERS: We're pretty much writing everything from scratch. We had our own collision system up until this current project, and now we're using a licensed one. We're still not sure if that was the right choice, but once you make a choice, you should keep going with it. We're using some parts of other stuff. We're using audio middleware and physics middleware, but pretty much everything else we're doing is from scratch.

I don't know if it's just because we're crazy, or because it's a little bit easier in the MMO space. The projects are so big that it almost seems as though the graphics engine—while that's a major component—and the stuff that we want in there that's major is the stuff that doesn't come with things like Unreal anyhow. There's always the cost of getting something that somebody else did. How much is it going to cost to change it to what you want it to be? We're still doing everything from scratch.

HECKER: You're not doing everything from scratch, though. Physics is huge—it feeds back into your gameplay. It's not just sound.

ROGERS: I know physics is exciting to you, but the point is that we're using a physics engine because we thought it would be neat. But we're not currently using it for physics; we're using it for collision, which is a system we already had. The system we had before, while not as technically advanced, was written specifically for the game that we were doing. So far, we haven't had any advantage from using this fancy, complicated piece of code.

HECKER: I wasn't trying to steer it toward physics—it just happened to be what you mentioned. It seems like there are different kinds of technologies. There are pure leaf technologies, like Miles Sounds System or something similar, where it's like, "Hey, I threw some audio data into this thing and out it goes!" It's output only, right? And then there's something like a physics system, whether it's collision or AI scripting or whatever, that has feedback loops back into the game.

It's way more complicated, even if you only use it for collision. You bite off this big chunk of work, and it changes your abstraction and flow control. It seems like there's a qualitative difference between those two things; stuff like output-only technologies is where we've been relatively successful. It seems as though on the things that have feedback loops, we've been way less successful.

ROGERS: It's almost like looking at how programmers reuse their own code. If you've got a nice function that does something simple like compare two strings or calculate a certain expression, everybody cuts and pastes that until the end of time, and you have perfect reuse on it. But the framework, the high-level thing at the top, is always very specific, and if you don't like exactly what it does, it's very difficult to adapt it to what you want to do.

Physics is maybe not a good argument to use to define what the problem is because it seems to be a very special case. It's a very complicated problem that most people don't have the training or interest to solve, and it has become a requirement of modern games. Whether it's a good idea to integrate or not, it's not a business decision that you get to make.

HECKER: The question of reuse is interesting to me, because it speaks to this thing we talked about earlier about C versus C++ and level of abstraction and iteration cycles and that sort of thing. It seems like reuse in the software industry has failed, especially on the C++-style module-level and object-level reuse. True or false?

ROGERS: True. I don't know if anyone expected it to work.

HECKER: There seemed to be a lot of articles about it 10 years ago. I wonder if it was ever possible. If it's not possible, then how do we scale from where we are, in levels of complexity? I think what makes games amazing to work on is this level of interactivity—it's what makes

it a new art form, potentially. To get that deep level of interactivity, you have interacting systems and a deep connectedness. That code just gets really complicated, and I don't know how we can get nonlinear productivity scaling there.

ROGERS: Maybe you don't! I think that there is this distinction that we make—the obvious example is low-level functions that people easily reuse. There are also layers of abstraction. Almost nobody programs directly to the hardware. It's very frequent now that people are using APIs to talk to very complicated hardware devices.

The most straightforward thing that has always worked in the software and probably even the hardware industry is that you build an abstraction for the system, and then you build an abstraction on top of that. Then, when you find out that an abstraction is a good, solid one, you can say, "Okay, this is mostly done and we can treat it as though it is reality, even though it's just hiding what's underneath it." When you get good, solid abstractions, that's when you can climb the hill to the next level. Trying to do that at the middle or the top of the tree has just failed over and over again. The engine licensing thing—where people are basically being given a structure that they can drill holes in and pop new parts in—while it's been shown to be really expensive to make that kind of integration, it's still cheaper than building everything yourself.

HECKER: The question is, does that put a ceiling on the level of innovation that you can do, in some sense?

BULKLEY: I think it consolidates the time you start to see new engines and new features, and it just takes longer and longer before that kind of stuff comes online. I don't see engines in a couple of years making the same kind of leap that they would have in a two-year period 10 years ago.

HECKER: So are they good for innovation or not?

BULKLEY: I don't think they're good for innovation, just because they do get so big and complicated that it's too difficult and too time-consuming to do

“ There are so many levels of innovation. For me, the most exciting stuff is happening in gameplay and the game itself. That's on a level that, to me, is separate from the engine. ”

—Stuart Denman



OptiTrack[™]
by NaturalPoint[®]

***YOUR OWN
MOTION CAPTURE
STUDIO
FOR \$5,000***

See it in action!

Visit booth #1621 at SIGGRAPH 2007

Package includes:



- ARENA[™] motion capture software
- Six OptiTrack FLEX:V100 cameras & accessories

Available Fall 2007.

Visit OptiTrack.com for complete details!

Pricing, package contents, and release date are subject to change.

Copyright 2007 © NaturalPoint Inc. OptiTrack, ARENA, NaturalPoint, trademarked slogans, and logos are the property of NaturalPoint. All other product names are trademarks or registered trademarks of their respective companies. All rights reserved.



that kind of stuff during a project. What you're waiting for is a separate R&D team from whom you can license, to spend all that time doing the R&D for you, and then later you can just jump on board.

DENMAN: There are so many levels of innovation. For me, the most exciting stuff is happening in gameplay and the game itself. That's on a level that, to me, is separate from the engine.

HECKER: Is it really? If you buy something, you buy a big framework-style engine, right? Does it make it easier or harder or does it not affect the gameplay innovation? To me, it's affected massively.

ROGERS: I think the whole conversation is too generic. If you get an engine, it allows you to do some things much more quickly, and it makes other things way too complicated because it already did them, so you have to decide where you want to innovate.

DENMAN: So what are you saying, Chris? Are you saying that just because you license Unreal, you can't create a checkers game?

HECKER: No. What I'm saying is that it depends on how deep the interactivity or innovation you're trying to do is. If you're trying to do a game that is in an existing genre and then add feature-level IP—basically adding a new thing that makes it part of the genre—it's clearly the way to go. You've taken an engine that is in that genre, and you threw in a bunch of levels and added some code, and you're gold. The question is, do you get new genres?

DENMAN: The point of an engine is that I can come in and say, "I'm Chris Hecker, I'm going to create the most innovative animation system in the entire industry. And I'm going to do it inside of, say, Unreal." Then you can ideally just rip out Unreal's animation system and put in your own.

KEITH: Your effort is going to be a lot more expensive than just doing it yourself with your own system. I've seen the homegrown engines over the life of a company, like 10 years, where legacy code is just a drag on innovation itself. People get so stuck to the idea of reuse or not breaking a system that might be a little bit fragile that it really reduces the level of innovation each year.

DENMAN: If you have a car, let's say, and your transmission breaks down, you're just going to be working on that part. It's the same way. Suppose I want to put Chris Hecker's animation system into some well-designed and modular engine and the idea is that the audio system is going to keep functioning and so are the art pipelines, engine framework, and graphics. Our artists and other team members can still be working on other parts of the game.

KEITH: I would love to see a system like that! There's so much cross-dependency across all systems; like audio calls in the animation system ... There are companies that have tried to replace the Unreal 3 physics engine, and they're replacing these little tendrils of control that are throughout the entire engine. It's not like a transmission. You can't pull four bolts off. The old part goes out, and the new one goes in.

ROGERS: There's an interesting point, where it's about the narrowness and how well-defined it is. It's like what I said before about the threading with the FIFOs. You try to have a really narrow interface between two things so you can see what problems are coming in. When you get a big, complicated system, you want to avoid as much as possible the inter-moving parts that have to communicate with each other. That's what makes the system interesting, but where those things happen, you want to have as well-defined an API as possible.

In some future world where everything is great, wouldn't it be nice if we could have this animation system that had this well-defined API, where you send it certain things and other certain things happen, and if some other system changes and starts sending it bad commands, you can easily track that down? I think that as software gets bigger, we're going to put more and more energy into defining these things, documenting them, and making sure they work correctly.

BULKLEY: I want to jump back a bit to the point about refusal to throw out code and it being a drag on innovation. There's a great Joel on Software [weblog by Joel Spolsky] article where he lists one of these things that you should never do—Netscape had a functioning thing, and it was working fine,

and they decided to throw it out and start from scratch. The problem with that is that you don't usually end up with something that's actually better. It just seems better when you're writing it because you're in the midst of it and you can understand it, but you come back a month later and you realize that you've made the same mistakes over again, you're debugging the same exact things that you've fixed before.

HECKER: There's also a little bit of an evolution versus revolution thing going on here. You can get stuck at local maxima, and the question is: Can you always iterate your way to the new spot you need to hit? It's like, "I'm going to change as little as possible in my current code base as I move it across the landscape," or "Do I need to tear it down every once in awhile?" I think animation systems are a bad example because they are getting close to being solid. But take AI middleware; would that ever work? Is the AI too essential to your game being different enough to work as middleware?

DENMAN: It's fascinating because to me, the way you architect your AI system is totally dependent on the users, designers, and programmers that you have using the systems. Different psychological patterns, intelligence levels, and all these different factors can come into play. You mention the visual versus the scripting language. It's an example of something that's dependent more on the people working on it than anything. You have this sort of psychological human element that comes into play, and that's why I don't think something like AI middleware could ever really be successful.

BULKLEY: I actually feel the same way about animation. I know that you mentioned that as one of the problems that are more or less solved, but ...

HECKER: I don't think it's solved, but it's easier than AI. Audio is the easiest, and then graphics and maybe physics on the ramp. Is that ramp just a linear ramp, and will we eventually get to the place where AI is solved too? Or is it too fundamental to the creativity of games? ❖

The editors acknowledge Martin Ecker of High Moon for his assistance with technical verification.



Canadian-born Mark Rein is Vice President and Co-Founder of Epic Games based in Cary, North Carolina. Epic's Unreal Engine 3 has won Game Developer Magazine's Frontline Award for Best Game Engine for the past three years and Epic was recently awarded Best Studio at the Spike TV Video Game Awards. Epic recently shipped Gears of War, which won Gamespot's overall Game of the Year and broke Xbox 360 sales records. Epic is currently working on the Unreal Tournament 3 for publisher Midway.

Upcoming Epic Attended Events:

Edinburgh Interactive Festival
Edinburgh, UK
August 15-17, 2007

Games Convention
Leipzig, Germany
August 22-24, 2007

GDC China
Shanghai, China
August 27-29, 2007

Tokyo Game Show
Makuhari Messe, Japan
September 20-22, 2007

Please email:
mrein@epicgames.com
for appointments.



Unreal® Technology News

by Mark Rein, Epic Games, Inc.

ANOTHER FANTASTIC E3 FOR UNREAL ENGINE 3

Here's a look at key Unreal Engine 3 announcements and milestones from E3 this year...

PLAYSTATION 3 GETS OPTIMIZED UE3

Sony Computer Entertainment (SCE) and Epic Games announced a multi-stage agreement whereby will work closely together to optimize UE3 for PS3 development to assist developers using the engine to accelerate game creation and maximize performance. Already underway, this optimization currently affects several developers using the Unreal Engine to create 20 games already in the works and with many more to come.

UNREAL TOURNAMENT 3 EXCLUSIVE ON PS3 THIS YEAR AND REDEFINES USER-CREATED CONTENT FOR CONSOLES

Also announced at the Sony press briefing was that Unreal Tournament 3 is a console exclusive on PS3 for this year and that, thanks to the open nature of the PLAYSTATION Network, users will, for the first time ever on a home console, be able to take the robust user created content (or "mods") created using Unreal Tournament 3's UE3 toolset on PC and play them on PS3. With great performance, and the ability to support user-created mods, Unreal Tournament 3 on PS3 will be everything that gamers have come to expect from the Unreal Tournament franchise. One news site reported that "you can definitely tell that the engine has been well optimized to run spectacularly well on the PS3". This is great news for UE3 licensees on PS3 especially considering that we'll be continuing to optimize PS3 performance leading up to the release of Unreal Tournament 3 on PS3 in November 2007. Windows XP/Vista, Macintosh and Linux will also ship this fall.

Unreal Tournament 3 - coming to PC and PS3 in November 2007



Unreal Tournament 3 - coming to PC and PS3 in November 2007

GEARS OF WAR COMES TO WINDOWS

That Gears on Windows, the #1 selling game on Xbox 360, blew people away with great smoothness and performance shouldn't normally come as a surprise. However the big surprise was seeing it on PC at E3. To show off the upcoming Windows version of the game, and how well our UE3 Windows optimizations are coming along, we staged multiplayer games of Gears on PCs at 1920x1200 resolution. Many users were amazed at the 60 frames per second performance

on great hardware provided by our partners Dell and NVIDIA. Equal effort is being put toward making the game, and engine, run on lower-end PCs as well. Gears of War will ship this fall for Windows XP and Vista.

UNREAL ENGINE 3 TO SUPPORT GAMES FOR WINDOWS LIVE

Microsoft also announced during their E3 press briefing that Unreal Engine 3 will support Games for Windows LIVE. According to Microsoft "this landmark announcement will make it easier than ever for developers to take advantage of Microsoft's online gaming network, through Epic's next generation technology."

KUJU HAS THE CHEMISTRY FOR UE3

"Chemistry" is one of five specialist studios formed by Kuju Entertainment Ltd. Formerly known as 'Kuju Sheffield' (founded in 2002), Chemistry recently announced that they're going to use UE3 exclusively for this console generation.

Specializing in UE3 allows them to get continuously more proficient at using it and develop increasingly polished games over time. We know this strategy very well because it is exactly what we're doing here at Epic. Like Epic they can take advantage of having UE3 as their common technology base with multiple teams sharing resources and

expertise to built world-class games in the most efficient fashion. UE3's mature toolset, and cross-platform capabilities should help ensure their success

Simeon Pashley, Studio Head, explained "we've decided to specialize in Unreal Engine as it's clearly an awesome next-gen toolset that frees our staff to focus on bringing their creative talents to bear without worrying about the low-level nuts and bolts. It's also perfect for our future plans as it allows us to look forward without worrying about technology stability". Chemistry has two Unreal Engine 3 based games currently in development.



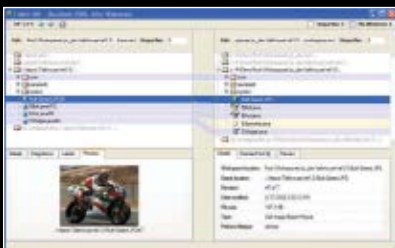
For UE3 licensing inquiries email:
licensing@epicgames.com

For Epic job information visit:
www.epicgames.com/epic_jobs.html

WWW.EPICGAMES.COM



Introducing Folder Diff, a productivity feature of Perforce SCM.



Perforce Folder Diff

Folder Diff is an interactive, side-by-side display for comparing the state of any two groups of files.

Use Folder Diff to quickly determine the differences between files in folders, branches, labels, or your local disk. This is especially useful when performing complex code merges.

And when you've been working offline, Folder Diff makes it a snap to reconcile and catch up with the Perforce Server when you get back online.

Folder Diff is just one of the many productivity tools that come with the Perforce SCM System.

THE CODE/ART DIVIDE



HOW TECHNICAL ARTISTS BRIDGE THE GAP

» **AS GAME DEVELOPMENT BECOMES MORE COMPLEX,** with bloated budgets and team sizes doubling over last generation development, the need to bridge the gap between art and programming has never been more pressing. It's already passé to rely on programmers to develop art pipelines and understand those needs. Game development today needs to be far more efficient, able to produce high-quality triple-A titles with team sizes comparable to those seen in the last hardware generation.

The technical artist is a new concept and role in the game industry that is starting to take hold. Every company has a different idea of what a technical artist's roles and responsibilities are. However, to really maximize the development process, a company must integrate the technical artist to the fullest capacity.

What follows is a blueprint for how we've integrated technical artists into our game development process at

Volition, which will hopefully give you ideas as to how your studio can do the same.

CASE IN POINT

During development of SAINTS ROW for the Xbox 360, I was responsible for designing and developing many of the tools and pipelines used for getting art assets into the game. But before Volition had any technical artists, it was up to the programmers to design and develop these systems for the artists to use.

Like any studio that doesn't employ technical artists, by and large programmers are the ones who develop and support the art pipelines. Generally, an artist will submit a request, and at some point in the future they are presented a tool to use. Most of the time these tools are not easy to use from an artist's perspective, nor is the workflow clear. Moreover, diagnosing problems with the

JASON HAYES is a technical art director at Volition, working on an unannounced game. He was senior technical artist on SAINTS ROW for the Xbox 360 and has been in the industry for 10 years. Contact him at jhayes@gdmag.com.

THE CODE/ART DIVIDE

tools can become problematic because getting programmer time to address them is typically difficult, especially near the end of a project cycle.

Integrating technical artists into a studio frees up the programmers from being solely responsible for the development and maintenance of the game's tools and pipelines. While programmers still have a hand in the design (and sometimes implementation) of those tools and pipelines, the technical artist is the driving force behind them and is looking out for the best interests of both parties. This allows programmers to focus more on developing game code, and artists to focus on making the best-looking content they can with easy-to-use tools and workflows.

THE ART OF DIPLOMACY

The following scenario, which took place near the end of development for SAINTS ROW, illustrates the importance of utilizing a technical artist.

Our game was having some serious frame rate problems, especially during nighttime gameplay. This happened for a number of reasons, but primarily because we were developing the game long before any hardware was available, and it was a new type of genre for the studio.

One of the many causes for the frame rate problems was our liberal use of per-pixel dynamic lighting. Since we were running low on time, many people on the programming side felt that it would be best to turn off the dynamic lighting at night and fake it with effects. On the art side, there was a desire to keep it because the lighting gave the night scenes a much better sense of believability and richness. All things being equal, programming would have won that fight because it's better to ship a game with stable frame rate than not.

Because of my knowledge of the engine and its capabilities and limitations, I proposed that we develop a hybrid solution to the problem. Dynamic lights would remain on at certain distances around the player, while further out, effects gave the impression of a much more well lit city without paying the GPU and CPU costs associated with the dynamic lighting. This

along with other optimizations improved the frame rate during nighttime gameplay dramatically, which kept the programmers happy. And visually, we still retained the believability and richness the artists wanted.

Cases like the one I described above happen often, sometimes daily. It's important to have an experienced technical artist between the two disciplines to negotiate what's best and important for not only each of the respective departments, but the product as a whole.

PIPELINE AND SYSTEMS ARCHITECT

Generally speaking, the technical artist should be able to design and develop all art pipelines necessary for the game. In this sense, part of the technical artist's role is to be a pipeline and systems architect. At Volition, this works on a few different levels.

Depending on the system, we work with the programming and art departments to determine what works best for both parties

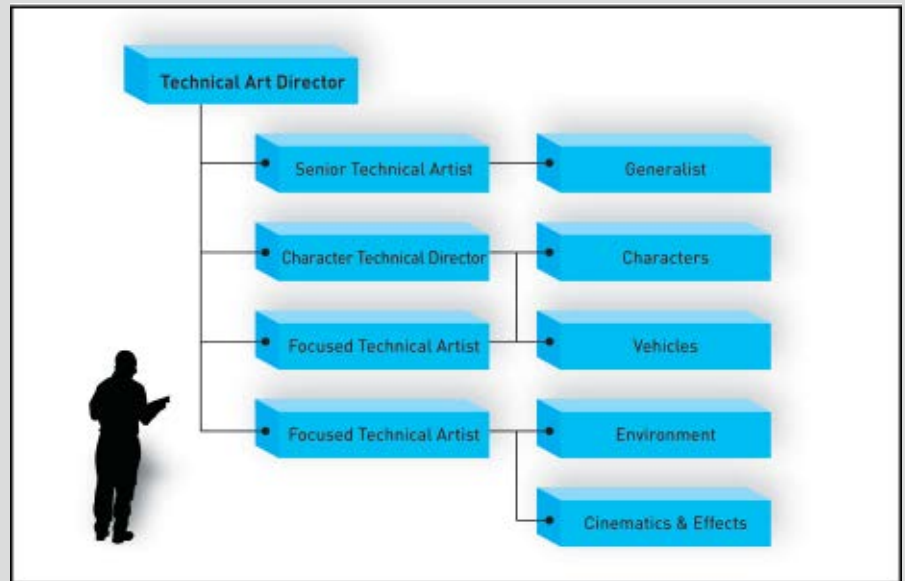


FIGURE 1 Volition's typical technical art team structure is shown.

and try to reach common ground. At this level, you could say that we act as negotiators between the technical- and content-oriented disciplines.

However, designing critical game systems requires technical artists who have intimate knowledge of both the game engine and development hardware, such as the Xbox 360 or PlayStation 3. The degree of knowledge necessary is such that if the technical artist isn't experienced enough or hasn't made due diligence a priority, pipelines can quickly take a turn for the



worse. More often than not, early mistakes are felt in the middle of production, or even later in the development cycle.

Not only do technical artists design and spec-out these systems (in coordination with other disciplines), they are the ones driving and championing the changes. Because of their intimacy with such systems, it makes them the primary source of information when it comes to how things work and fixing bugs in tools written to support the pipeline.

It is important to note that at Volition our technical artists do not design code structure for the programmers. This level of granularity is neither our job nor our area of expertise. In contrast, we work with them at a higher level to develop the best way to get requested features from Point A (content creation) to Point B (the game) and everything in between.

CONTENT BUDGETS AND GUIDELINES

The technical artist is the driving force behind setting budgets and developing technical guidelines for creating art content. It's important for the technical artist to remember that while these requirements help to serve game performance, they must also be balanced such that they allow for a high degree of visual quality.

Art should be authored in a way that doesn't hinder performance of the game. Instead, it should be created to take advantage of the game engine and hardware strengths. To meet these goals, the technical artist works closely with the team's rendering programmers.

TOOLS PROGRAMMER

A technical artist should be flexible enough that she or he can develop tools for the art pipeline without any assistance. Typically, programmers do not wish to write these tools and this is where a technical artist can fill that void.

The level at which tools are developed varies. Most technical artists, by today's standards, come from an artistic background and favor the use of dynamic scripting languages such as MaxScript or Mel.

As their interest and experience level grows, some technical artists grow into more lower-level languages such as C# and C++. This gives them greater ability to write compiled plug-ins, such as exporters, and tools outside the content creation packages.

SPECIALIZED DISCIPLINES

If there ever were a difficult position to fill, it's that of the technical artist. Technical artists were once content artists who taught themselves to write scripts out of necessity and had a natural interest in the technical side of the craft. At Volition, we break down these roles into core areas of game development. See Figure 1 for an example of the typical structure for our technical art teams.

Technical art director. The technical art director is at the same level as the art director in our typical team structure. This person is responsible for coordinating the technical art team, prioritizing features, identifying and assessing project risks, and scheduling and designing critical tools. In addition, the

technical art director also designs and implements game systems and pipelines, creates guidelines and budgets for art content creation, and makes sure that the game's rendering performance is running optimally, while working with the art director to maintain a high degree of visual quality.

Generalists. What we call generalists are typically the senior technical artists who can drive any system in the game. They have a wide range of experience to pull from and are typically the critical go-to guys.

Character technical director. The character technical director is responsible for setting up the character skeletons, rigging, identifying and assessing motion capture and animation needs, scheduling and coordinating the animators, and developing or designing tools and pipelines to support the game's character systems.

Senior technical artist. The senior technical artist is primarily responsible for the design and implementation of larger and more critical game systems and pipelines. She or he is also partly responsible for ensuring that content is being created in an optimal fashion for not only rendering performance, but high visual quality as well.

Focused technical artist. Focused technical artists are typically entry- to mid-level technical artists. They focus primarily on specific areas of the game, such as environment art or character art. These focused technical artists take the point position for their particular art department and get approvals through the generalist or technical art director types. They provide direct support and develop any needed tools and pipelines necessary for their respective department(s).

TECHNICAL SUPPORT

Another major area of responsibility for technical artists is to provide technical support to the art team. This includes tasks such as diagnosing a problem an artist is dealing with in the content creation package (3ds Max, Maya) or other in-house proprietary tools.

Filling in the support needs of artists can be a time-consuming process. To speed up the process for technical artists, we require that all requests be sent through email in the form of a descriptive explanation of the error or problem, along with attached screenshots.

We do this for several reasons. At Volition, technical artists support a vast number of artists, including any who are outsourced. It's critical that the response be focused to reduce the amount of time spent on the request and get the artists back up and running so they can continue to work with as little interruption as possible.

SCHEDULING

A lot of what a technical art team does (depending on how far the team is in the development process) affects scheduling, both their own and the project's. As individuals, technical artists frequently switch between developing tools and fighting fires, often at a moment's notice. And as multi-disciplined team

THE CODE/ART DIVIDE

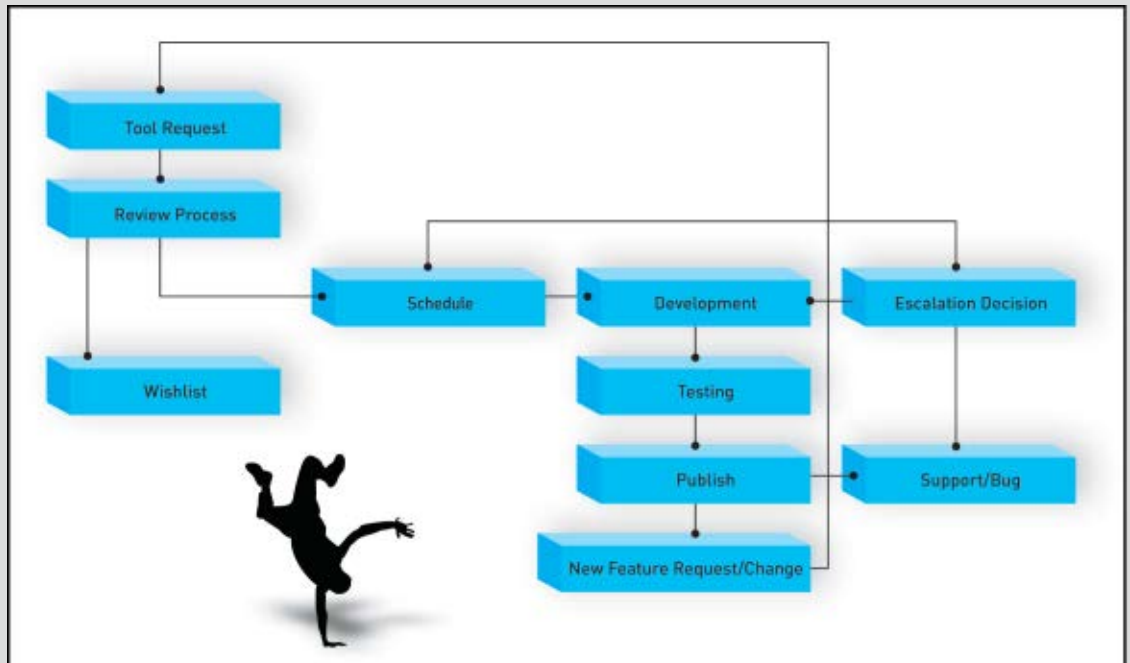


FIGURE 2 Volition's flowchart accounts for feature requests as well as a change management plan.

members, they are required to be in many other departmental meetings, some of which come up unexpectedly. If management accepts that the discipline is difficult to schedule and maintains an open mind, scheduling technical artists can be manageable.

Here are a few approaches we've found that work well in creating technical artists' schedules.

Early identification of needs. At Volition, we allow anyone in the studio to submit ideas for a feature or new tool to the project via email aliases. These items are reviewed by all project leads and dependencies are identified at this stage.

When evaluating a request, we use a five-point rating scale, with 1 signifying "must have." We make it a priority to fill all level 1 and 2 requests, while items with priority levels 3 through 5 are reviewed later to fill out empty spots in schedules.

Due diligence. Every tool or feature request goes through a three-phase process: investigation, implementation and documentation.

Investigation is used to identify risks of the request. Too often, risks are not accounted for in the schedule; by identifying them, you impress the importance of building solid tools.

Documentation is often glossed over. Good documentation ensures that anyone using the tool, no matter their technical ability, will use it properly.

Schedule support time. This is to accommodate the roller coaster frequency of support calls. From our experience, we have found support time typically ramps up quickly the closer

you get to the end of a milestone or deadline.

Schedule buffer time. Even with the scheduled support time, things inevitably crop up that can't be foreseen.

Change management. Implementing a solid change management plan for tools and feature requests is essential to keeping to the schedule. Too often, features are requested for existing tools that may seem minor to implement, but added all together they further complicate the already difficult problem of scheduling tech art. See Figure 2 for a flowchart that illustrates how we have implemented our feature request and change management plan.

IMPLEMENTATION

How many technical artists should a company have? Over the past few years at Volition, we've found a need for three or four per project with a team size of roughly 80 to 90 people.

We structure the team such that there is the lead, who is the technical art director, then at least one senior technical artist, and a character technical director. The others are more focused technical artists who are assigned to specifically dial in on certain areas of the game.

While finding the right person to fill this role is difficult, it should not be overlooked in today's competitive and high cost environment. If your studio has no technical artists at all, or has some that aren't being used to their full potential, I encourage you to take another look. You'll be glad you did. ❖

Outdated Character Tools Have Control of You?

- Author Behaviors Interactively
- Plug in custom procedural nodes
- Leverage the full power of Havok Physics and Animation

Havok Behavior Gives You
Complete Character Control

havok
www.havok.com



FINAL FANTASY XII

THE FINAL FANTASY FRANCHISE IS ONE OF THE CORE BRANDS IN the Square Enix lineup. When we set forth to develop the twelfth installment, there were three titles that could be considered its immediate predecessor: FINAL FANTASY IX, X, and XI, all of which were announced simultaneously at a press event held in January 2000.

Each of these three games represented a new chapter in game development for the franchise. One was the last FINAL FANTASY for the PlayStation. One was the first FINAL FANTASY for the PlayStation 2. And the last was Square Enix's first network game, which would become the core of the PlayOnline network service.

Around the time we started developing Final FANTASY XII, FINAL FANTASY X got a sequel in FINAL FANTASY X-2, and the team for that title remained the same. Also, the service for FINAL FANTASY XI garnered far greater success than we had expected, so that team kept working as well.

Since FINAL FANTASY X and XI were being developed, the teams that had been working on FINAL FANTASY TACTICS or VAGRANT STORY came together to create FINAL FANTASY XII, introducing a third FINAL FANTASY console and PC team.

The individuals on this team (myself included) had previous experience developing action/real-time-based battle systems for games, which creates a connection between the fields and battles. We decided to make use of our prior knowledge and integrate an action/real-time-based battle system into the FINAL FANTASY franchise, which was a significant challenge. The previous releases of FINAL FANTASY used a random encounter-based battle system.

Thus began the development for this project. We would soon face various complications that were completely beyond the scope of our imaginations. The entire process was a tremendous learning experience.

WHAT WENT RIGHT

1 UNIQUE AND FRESH DEVELOPMENT TEAM. While it's true that we had to create a third development team for the FINAL FANTASY franchise, in reality, all the teams were already evolving, and they rarely comprised the same people.

Still, this new team was unique from the others in that our team culture had been defined and established through our

GAME DATA

DEVELOPER
Square Enix

PUBLISHER
Square Enix

PLATFORM
PlayStation 2

RELEASE DATE
October 31, 2006

SOFTWARE USED
Autodesk Maya,
Softimage XSI,
OPTiX iMageStudio

MONSTERS
250

BOSSSES
30

NPCs
1,000

TAKU MURATA is general manager of research and development for Square Enix Co., Ltd. He was supervisor for FINAL FANTASY XII, and was instrumental in the development of the company's PlayOnline network service. Send comments about this article to editors@gdmag.com.

previous and varied work experiences. Many of the staff members had never been a part of a project the size of a FINAL FANTASY game before in terms of financial resources, time, and staff.

Having new blood is something that “went right” because the team was able to look at the development process with new eyes and thus use new techniques that I will divulge throughout this article.

2 IN-HOUSE TOOLS. Using in-house tools was the theme of a talk I gave at the 2007 Game Developers Conference. Our various in-house authoring tools, coupled with commercial digital content creation tools, such as Photoshop, Maya, and Softimage XSI, created an environment in which we could use trial-and-error tactics with the new tools while also increasing productivity by using the ones we already knew well.

It was especially helpful for us that the in-house tools enabled real-time previews using the game’s rendering engine. Due to the overwhelmingly positive experiences developing FINAL FANTASY XII, our company started to strategically incorporate tools in other development processes companywide.

3 DATA MANAGEMENT. The management of data from an immense number of resources—especially the progression from in-house order placement to completion—was an important part of the development process. To monitor the data, we prepared a web site solely for the project where we could collect information, systemize the overall development process, and ultimately make management as smooth as possible.

On past projects smaller than medium-sized, we had addressed data management concerns by throwing manpower at them. For Final Fantasy XII, however, we developed a tool that allowed users to perform constant status checks, including a check on the progress of closely related divisions.

Everyone who had been familiar with the management of medium-sized projects was grateful for a tool that could simply regulate communication more effectively. Nowadays, basic software configuration management systems can ease the pains of data management, but at the time we were pleased with our ability to come up with an original in-house tool.

4 THE DELEGATION OF AUTHORITY AND PLACEMENT OF HUMAN RESOURCES. The FINAL FANTASY franchise has given Square Enix the opportunity to discover human resources on a large scale. All too often in large-scale projects, those in leading positions are plagued by blind spots, which could easily become bottlenecks.

To avoid this problem, several staff members were given different positions of authority that required them to not only perform to the best of their particular skills, but also go above and beyond the call of duty in a number of new positions and duties. Many staff members emerged as top players during this experience and ultimately moved forward in their careers.

Another advantage of large-scale projects is that the staff members’ personal network becomes larger, and because more people enter the director’s field of vision, people can expand their skills to different areas.

As a result, our status as “third team” became more stable, and the developers—the majority of whom are graphics designers—improved their skills dramatically.

5 STABILIZING THE GAME SYSTEM AT AN EARLY STAGE. The programmers made it their priority to stabilize the basic game system at an early stage of development. Q/A started play testing at an early stage as well, partly due to the fact that a playable ROM had to be exhibited at E3 2004. Moreover, a stabilization of the overall game system was secured early on by strictly maintaining the policy of removing any itemized reactions from the system toward minute specifications, something that commonly occurred in previous projects.

For the Q/A team, a stable game system meant that they could focus more on play testing and that all their work was relevant.



WHAT WENT WRONG

Looking back at the development process of FINAL FANTASY XII, the problems that I am aware of can be roughly divided into two groups: scheduling and over-ambitiousness. First, we should have realized sooner than we did [like when it became evident in our schedule management] that the project would exceed our forecasted scale. Second, several elements of the project had gotten out of control; the number of ideas to implement had grown, but resources and communication had not.

At the risk of making this article somewhat lopsided, I would like to place my focus on “what went wrong” instead of “what went right” because the process of overcoming the “wrongs” was vital for me as general manager of the research and development division.

As in life, in game development one learns more by going through rough times than times that pass without much conflict. Working through the difficult times essentially provided a core element in launching the R&D division. And through these lessons, we actually created a new structure within our company.

1 UNPREDICTABILITY. One of the major challenges in developing FINAL FANTASY XII was integrating our new action/real-time-based battling system. It raised the level of difficulty for development exponentially.

Within the company, we already had capability and experience developing both action/real-time-based battling systems and traditional FINAL FANTASY titles. But when the new team attempted



to blend these two elements, they already had a preconceived notion of their own formula for success. At the same time, each team member had his or her own image of the FINAL FANTASY franchise. What we should have done was take these notions into account and create a test model and prototype in order to assess the risks and consequences. What ultimately stood before us was an unpredictable outcome in too many areas.

We couldn't decide where to split the workload within related divisions or how partial alterations to the game would affect existing standards. As a result, the staff found themselves in a state of endless debate and adjustments. Their doubts directly affected the development schedule, which had a serious effect on the overall development and caused delays in everything from the determination of specs, to snap decisions within production, to the game's balance, to the formation of a Q/A team.

2 INNER PRESSURE OF DEVELOPMENT POWER. The internal pressure we all felt in our work environment at the time was comparable to a pressure cooker filled to the brim and set on high. This is a problem that Square Enix as an entity was destined to encounter, and it was especially prominent in the development of graphic resources.

By constantly reinventing itself, Square Enix has heightened the skills and motivation of its staff. The company has achieved world-class standing in its visuals and audio. Moreover, the number of those involved in development is

immense, and the amount of ideas, techniques, and resources they create is massive.

During the development of FINAL FANTASY XII, the pressure to succeed was at such a high point that we were on the brink of losing control during even the slightest misunderstanding. What



happened was our team was given the freedom to make changes at various stages of development, but the adverse effect of this freedom was miscommunication, confusion, and



Accelerate up to x20:

- ✓ Visual Studio builds
- ✓ Make-Based Builds
- ✓ Data Builds
- ✓ R&D Scripts
- ✓ Custom Applications

IncrediBuild is an easy-to-use platform for accelerating Windows-based processes using advanced Grid Computing technology.



Download **FREE**, fully-functional 30-day trial:

<http://www.xoreax.com>

FORK PARTICLE

Particle System Middleware

Fire Up Your Game!

Artist friendly authoring tool
Quick and easy integration
Fast production pipeline
Runtime engine SDK

"I highly recommend Fork Particle, as an excellent tool for any effects team who desire the freedom to create spectacular and advanced effects quickly and easily."
Joe Jobst, Lead Effects Artist

Evaluation at:
www.forkparticle.com
sales@forkparticle.com

© 2007 Fork Particle Inc. Fork Particle logo is a trademark of Fork Particle Inc. The Mad Dog logo is a trademark of Mad Dog Software, LLC.

disorder. How work was to be distributed was also often ambiguous, which contributed to the problem.

3 LATENCY. Using in-house tools allowed the artists to use trial-and-error techniques in graphics pre-visualization, but it did not eliminate all the problems in the development environment.

Because the build process took so long and latency issues were so extreme, the content pipeline became long, and at times we were in a situation in which we had to limit the trial-and-error period on the design side. We couldn't afford any mistakes in order placement, and we couldn't try out a new idea just for fun, which affected the detailed elements that go into game design.

Since then, the system has already been improved via a reconstruction of the content pipeline and support from a distributed build environment, but it was a huge obstacle while we were working on FINAL FANTASY XII.

4 SAME OLD THING. Since I had developed a mid-sized project prior to joining the FINAL FANTASY XII team, I initially planned to use the same methods that I already knew. As a team, and as an individual, I felt that we could get through the project with this formula. The fact is I had become too dependent on human communication.

Considering the scope that FINAL FANTASY XII was aiming for, we needed a more systematic form of communication for data management issues. Luckily, we realized that there's a communication breakdown that is born from such an inadequate system. As the development team expanded, we renewed our method of large-scale data management, and the communication problem was solved before it could cause great harm. The fact that we altered our style according to the change in scale—in effect, throwing away the traditional ways and establishing a system to manage a large amount of data—was what enabled us to succeed.

5 THE DIFFICULTY OF Q/A. As I mentioned in the "What Went Right" section, we were able to stabilize the system at an early stage. However, toward the end of development, the Q/A team faced huge difficulties. Usually, because of its complexity and volume, the debugging and testing phase is organized with extreme care for a FINAL FANTASY game. However, for this project, the system's stability caused a slack in the predictions.

We miscalculated how long it would take to accommodate the exponential rise in volume from a game designed with an action/real-time battling system. Even with prior experience under our belts on the game VAGRANT STORY (which used a similar battle system), we still had no idea what to expect with FINAL FANTASY XII.

Furthermore, internal pressure in the development team caused more two

problems. First, the team became obsessed with corrections and revisions. The revisions created unforeseen bugs leaving the Q/A team to struggle between whether they should meet quality demands or the schedule.



The second problem was adjusting the difficulty level. The initial adjustments from the development side showed that some areas were too difficult for the FINAL FANTASY franchise, which attracts many casual players. We collected and analyzed the play-testers' first impressions in order to readjust the game balance. But in the final stages, we didn't have enough time for debugging. Our stress multiplied when we remembered that we couldn't hire more Q/A testers since it was close to the end of the fiscal year. The only reason we got through those tough times was because of the Q/A team's persistent efforts.

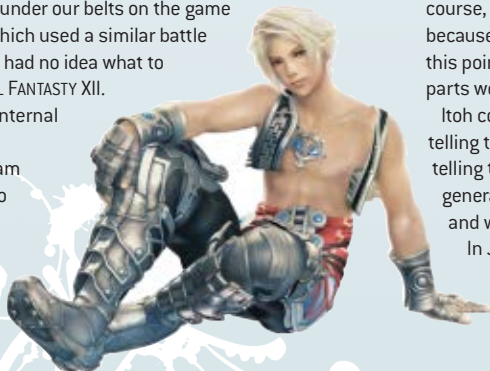
THE SURPRISING WATER

The final stages of FINAL FANTASY XII's development were a series of battles against the pressure from within. Actually, in order to write this article, we assembled all the key players together for a postmortem meeting. There, game design director Hiroyuki Itoh shared the following story.

In order to release his development team from the state of inflation that it had fallen into, Itoh told the planners under him to cease working on the quests that were lagging in production. He told them to throw it all away, even though the team was approaching the end of the scheduled development process. Of course, he was able to initiate this sort of shock treatment because he knew that there were enough completed quests at this point for the game to be complete, and he also knew which parts were burdening the staff.

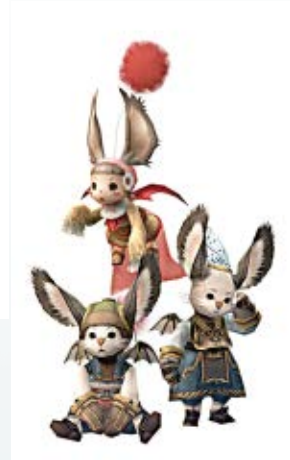
Itoh confided that he had doubts until the last minute about telling the team that it simply could not use all its resources, telling the staff to lighten their load. In Japan, game projects are generally very heavily planned on the front-end of the project, and we rarely cut features at the end.

In Japanese food culture, when boiling beans, the cook pours cold water called "surprising water" into the pot right before the boiling is complete. It is said that by decreasing the temperature of the water for a





The FINAL FANTASY XII key developers: back row, from left to right: Takashi Katano, main system and event programmer, Jun Akiyama, event director, Hiroshi Minagawa, director of visual design, Kazuhiro Kataoka, lead map system designer, and Hiroyuki Itoh, director of game design. Front row, from left to right: Yoshinori Tsuchida, real-time rendering programmer, Akihiko Yoshida, main character design, Taku Murata, supervisor, and Hiroaki Kato, project manager.



moment, a sort of resistance develops when one eats it. The bold direction to throw everything away had a similar effect on the staff members, and created an important turning point in preparation for the closing of development.

Of course, not all units could have taken such drastic measures. For example, the map production team was able to make a soft landing with the direction to add the final touches. Nevertheless, the danger presented by the pressure of development could only have been avoided by human response. How we will improve this condition is a major challenge that we

must address in the future. And when we are finally able to control this existing energy, we will finally be able to reach next-generation development.

EDUCATIONAL EXPERIENCE

At the time of this writing, more than 5 million units of FINAL FANTASY XII have shipped globally. All the team members take pride in the fact that they were able to overcome such insurmountable odds and produce a top-selling game. They're proud to have taken the FINAL FANTASY franchise to a new level by utilizing gameplay mechanics that were untouched until now.

I believe that the success we saw with this unique team was possible because we enforced a different work culture than had been seen previously in the company.

Currently, I hold a top position in Square Enix's research and development division. This division was created for the future expansion and complexity of our content, as well as to concentrate on the development of a technology platform and the fundamental technology to provide a fun gaming experience. The fact that I was able to take part in the development of a large-scale game like FINAL FANTASY XII has been very helpful in the operation of the organization and in deciding the direction that it will take. ✕



FINAL FANTASY XII utilized a new real-time action battle system.

PHOTOSHOP CS3 EXTENDED & 3DCONNEXION SPACENAVIGATOR

BY TOM CARROLL

THE NIGHT BEFORE THE CLOSE OF THE current iteration of The Platinum Studios Comic Book Challenge, I was up late finishing a page of comic book artwork for a submission called "Dread Naught." It involved a crew of zany aeronauts aloft in a high-tech zeppelin searching for a nefarious thief.

With only



hours left before the midnight deadline I had to choose to: 1) render a screen grab of the 3D model I had created of the airship so it could appear to be on approach to a distant spire in a fantasy environment I had sketched; 2) freehand draw the zeppelin into the frame and try to render it enough so it fit into the artwork; or 3) leave the airship out all together and go to bed.

I chose option 3, and I did so because the first two choices were fraught with peril, and anyone who has tried to take a screen grab of a 3D model and paste it into a piece of 2D artwork knows that getting it to work in the frame is about as easy as writing the Lord's Prayer on the head of a pin ... with a chainsaw.

If I had only had Adobe Photoshop CS3 and SpaceNavigator, my troubles would have been considerably less and I might have made the deadline with an airship in frame.

PHOTOSHOP CS3 EXTENDED

Adobe Photoshop was already the 800-pound gorilla of the image creation and editing field, but with the recent debut of Photoshop CS3 Extended, this jumbo gorilla just bulked up a bit, which should put the competition even more on edge.

Perhaps the most significant development for game developers is the ability to now drop 3D development models into new 3D layers where they can be moved, rotated, retextured, and lighted prior to becoming eye candy for such destinations as backgrounds, in-game textures, or marketing materials.

To test the possibilities, I first created a simple set of untextured cubes, exported them to a new format, and created a 3D layer that contained them. The results were very quick and quite fun to work with.

A co-worker suggested I import a piece of in-game geometry that I had been optimizing. The 6,500 triangle, flat-shaded asset transferred without a hitch,

and I could easily see the immense benefit of being able to manipulate an asset in Photoshop in 3D prior to squashing the geometry or flattening all the layers. The last test was to crop out my comic book environment and import the basic zeppelin model I had made within a 3D layer to begin testing how it might have looked had I gone that route at 10:15 the night the contest closed.

LIKE THE MONKEES, NOW I'M A BELIEVER

But how about the big picture for Adobe and its consumer base? Adobe Photoshop CS3 isn't cheap. Is this the first time that an 800-pound gorilla is simply too much gorilla for its own good?

In a word, yes.

While Adobe has opened the way for its hardcore elite to extend their capabilities into 3D modeling, animation, scientific applications, and medical imaging, CS3 will flat out overwhelm newbies. Anyone who stops at resizing or simple color correction (or who doesn't currently use Photoshop and is still reluctant to file taxes online) should go straight to Photoshop Elements at the retail aisle, which, in actuality, is precisely what Adobe wants, isn't it?

3DCONNEXION'S SPACENAVIGATOR

Anyone who wonders why 3Dconnexion's SpaceNavigator hasn't become the de facto standard for 3D joysticks should review the "crossing the chasm" model of product diffusion. Simply stated, the product developer or marketer focuses on one group of customers at a time, using each group as a base for marketing to the next. The most difficult step, however, is making the transition between visionaries and pragmatists. This is the chasm. Only after a product crosses that chasm does it have any chance of making it with a mass audience.

The SpaceNavigator is poised at the rim of the chasm.

Will it make it across?

3DCONNEXION SPACENAVIGATOR



3DCONNEXION

Headquarters
2010 N. First St., #510
San Jose, CA 95131
www.3dconnexion.com

PRICE
\$199

SYSTEM REQUIREMENTS

- Intel Pentium 4/III or AMD/Athlon processor based system
- 20MB free disk space for driver and plug-in installation
- USB 1.1 or 2.0.

Operating Systems Supported

- Windows XP SP2 x86

- Windows XP x64
- Windows Vista 32-bit or 64-bit
- Windows 2000
- UNIX (HP-UX 11, IBM AIX 5.2, Sun Solaris 8 or greater)
- Linux (Redhat Enterprise Linux WS 3, SuSE 9.0 or greater).

PROS

1. Once installed and functioning, the learning curve is extremely reasonable.
2. Complements mouse and keyboard in a very pragmatic way, especially when working on 3D

models or the 3D applications in Photoshop CS3.

3. Looks really cool.

CONS

1. Should allow users to define the install path, making it more intuitive to install drivers in non-standard places.
2. Annoying speed adjustments from application to application and lack of help documentation.
3. Could the serial number be any smaller on the USB cable tag?

OUR RATING SYSTEM :



EXCEPTIONAL



GREAT



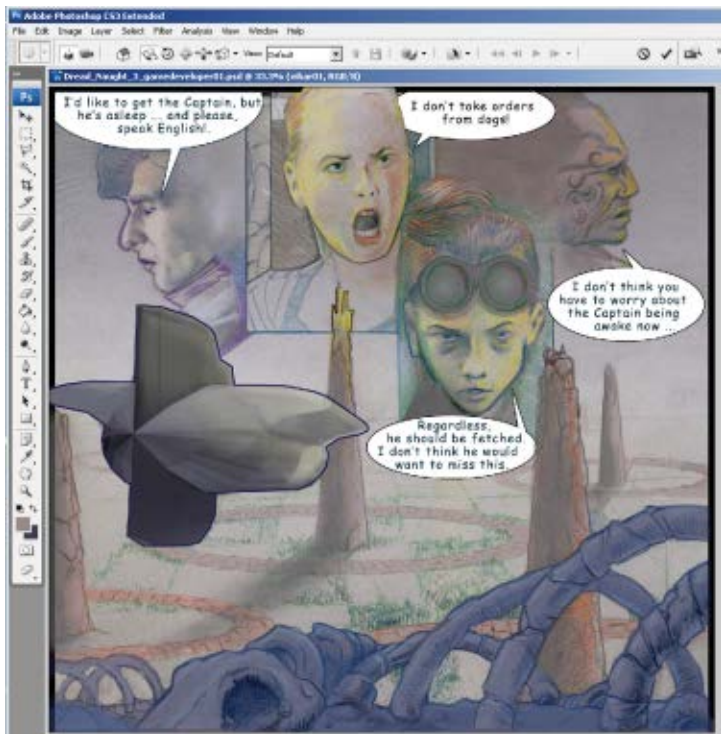
FAIR



POOR



UNFORTUNATE



Adobe's Photoshop CS3 Extended, which can place 3D creations in 2D artwork, is compatible with 3DConnexion's SpaceNavigator.

While I'm not of a mind to make bold predictions, I do know that it made the crossing for me, but not before narrowly passing over some speed bumps.

I found the SpaceNavigator to be a bit of a challenge to work with, not because the installation process wasn't quick and efficient, but, in fact, because it

was efficient—but not flexible, and flexibility is a key component in crossing the chasm.

I first installed the SpaceNavigator at my office because I wanted to use it with Maya. What I found was my Maya operated off a mirror drive; when the drivers installed on my C: drive, the Maya

on my I: drive didn't see them. When I looked for help, the readme files read "impossible to readme" because they had all the nuance and charm of a .perl user guide. For someone who is not an early adopter—that is, for someone like me who discovers how IT-challenged he is nearly every day—this was no fun at all.

The new Adobe Photoshop CS3 Extended installed on my C: drive, however, and it worked perfectly with the SpaceNavigator device right out of the box. Also, when I took both products home to my standalone system, they installed slick as a whistle. SpaceNavigator made 3D manipulation very fun, and I discovered that the learning curve was acceptably small for as much new functionality as was delivered.

3Dconnexion's peripherals have few, if any, competitors on the market. To cross the chasm to a broad user base that is blissfully unaware of the product, the company needs to grow its online and in-package support so it doesn't take a NASA expatriate to work through teething troubles. As soon as the company sees this solution, word of mouth alone may be enough to get more people to cross that \$100-a-pop chasm.

TOM CARROLL is a video game artist currently with Rockstar San Diego. He is also a contributor to *Twanks and Plankers*, an online comic gallery. Email him at tcarroll@gdmag.com.

CONTINUED ON PG 30

ADOBE PHOTOSHOP CS3 EXTENDED



ADOBE SYSTEMS INCORPORATED
345 Park Ave.
San Jose, CA 95110-2704
www.adobe.com

PRICE
\$999 or \$399 for upgrade

SYSTEM REQUIREMENTS

Windows

- Intel Pentium 4, Intel Centrino, Intel Xeon, or Intel Core Duo (or compatible) processor
- Microsoft Windows XP with

Service Pack 2 or Windows Vista Home Premium, Business, Ultimate, or Enterprise (certified for 32-bit editions)

- 512MB of RAM
- 64MB of video RAM
- 1GB of available hard disk space (additional free space required during installation)
- 1,024x768 monitor resolution with 16-bit video card
- DVD-ROM drive
- QuickTime 7 software required for multimedia features

- Internet or phone connection required for product activation.

Macintosh

- PowerPC G4 or G5 or multicore Intel processor
- Mac OS X v.10.4.8
- 512MB of RAM
- 64MB of video RAM
- 2GB of available hard disk space (additional free space required during installation)
- 1,024x768 monitor resolution with 16-bit video card
- DVD-ROM drive

- QuickTime 7 software required for multimedia features
- Internet or phone connection required for product activation services.

PROS

1. 3D objects in layers on a 2D document? Who would have thought?
2. CS3's improvements do improve workflow.
3. Massive amounts of third-party support, especially tutorials from www.lynda.com.

CONS

1. Still about 10 pounds of features in a 5 pound box, which keeps the newbie learning curve maxed out.
2. Fractured market now includes Adobe CS3, Photoshop Elements 5, and Corel Paint Shop Pro XI. Finding entry level is tougher than ever.
3. The huge installation is like a toad trying to swallow a python, but luckily for most of the world, it's a one-and-done operation.

CONTINUED FROM PG 29

product news

BRYCE 6

DAZ 3D

DAZ 3D, the 3D software and content creation company that makes Hexagon and DAZ Studio, has just released version 6 of the 3D software tool Bryce. The most recent version now sports HDRI-based lighting solutions. Other improvements include a Boolean mesh conversion tool that lets users export their creations into other 3D applications, and support for Intel-based Macs. Bryce 6, which includes DAZ Studio, can be purchased new in box for \$109.95, or via downloadable upgrade from Bryce 5 or 5.5 for \$39.95.

www.daz3d.com

BORIS CONTINUUM COMPLETE 5.0 FOR AVID AVX

BORIS FX

Boris FX, a company that specializes in integrated effects technology for video and film, has introduced its Boris Continuum Complete 5.0 for Avid AVX 2.0 architecture (BCC 5.0 for AVX). The tool, which became available this summer, is a desktop compositing plug-in package with

more than 180 filters for professional visual effects creation. Version 5.0 includes new on-screen controls for Avid AVX, multi-core processor or OGL accelerated filters (including the "Ken Burns" effect), and overall efficiency upgrades. BCC 5.0 for AVX is currently available for purchase through Boris FX's web site for \$895 (new) and \$295 (upgrade) for Avid Xpress Pro systems, and \$1,995 (new) and \$599 (upgrade) for Avid Media Composer and Symphony systems.

www.borissfx.com

GAMEBRYO 2.3

EMERGENT GAME TECHNOLOGIES

Emergent Game Technologies not long ago released a new version of its Gamebryo engine, the most notable changes to which include support for DirectX 10 and PhysX. Gamebryo 2.3 is optimized for development on Xbox 360, PlayStation 3, and PC. Other improvements include a new object-based rendering framework and additional support for multi-threaded development.

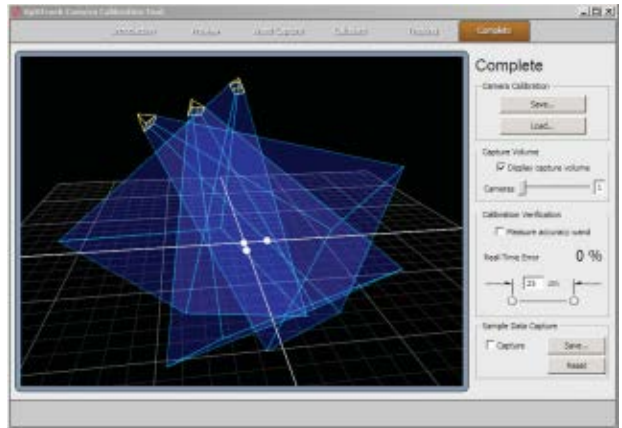
www.emergent.net

REALVIZ HDR IMAGING STUDIO

REALVIZ

Image processing software developer RealViz has released a new bundle of its products Stitcher Unlimited 5.6 (a photo and imaging stitching program, which was first offered in May), VTour 1.2 HDR (software that creates a 3D model environment from a photo reference), and Photomatrix Pro (a HDR program). Sold together as HDR Imaging Studio, the tools allow a user to build HDR panoramas, export them to another 3D package as environment maps, and add depth information to the HDR environment. You can purchase the Windows-only package via download from the company's web site for \$2,580.

www.realviz.com



Point Cloud from NaturalPoint is a new 3D marker-tracking system for motion capture.

POINT CLOUD AND FLEX:V100 MO-CAP CAMERA

NATURALPOINT

The Point Cloud toolkit, a recently released piece of software from NaturalPoint, is a real time, multi-camera, 3D marker-tracking system that can integrate with custom mo-cap solutions. Point Cloud is meant to facilitate the implementation and data retrieval of mo-cap sessions by making it more comprehensible to non-programmers. NaturalPoint is also now offering a new motion-capture camera, the FLEX:V100, which offers VGA resolution at 100fps, built-in strobed IR illumination, and onboard image processing. The company, which also owns the optical tracking products TrackIR, OptiTrack, and SmartNAV, is offering the FLEX:V100 (\$529) and Point Cloud Software (\$299) directly from its web site.

www.OptiTrack.com

HAPTIX

REACHIN TECHNOLOGIES

Sweden-based Reachin Technologies has released a new computer games-focused engine called Haptix, which the company says will let players feel details of objects in games, such as their texture and

weight; haptics, which is sometimes referred to as "force feedback" when used to describe hardware, is the science that deals with the sense of touch. Any games made using the Haptix engine will require that the player have a peripheral device, such as the Novint Falcon, which must be compatible with the game. In May, the company also joined the IGDA Partner Program, further supporting its place in the game community.

www.haptix.com

QUAD-CORE AMD OPTERON PROCESSOR

AMD

AMD has said it will begin shipping new energy-efficient, native x86 quad-core processors this month. According to AMD literature, the new processors, codenamed "Barcelona," are planned for shipment in both standard and low power versions at launch in August. The processors are designed to maximize performance per watt. For more information, visit AMD's web site.

www.amd.com

Send product news announcement to editors@gdmag.com.



**Don't tweeze
your nose hairs
on a rollercoaster.**

Do know your audience. Learn how with GameSpot Trax,
the industry's premier market-intelligence tool.



Find out more:
trax@gamespot.com

TOKYO GAME SHOW 2007



Link up, Reach out, To the World.

TOKYO **GAMESHOW** 2007

9/20·21·22·23 Venue: **Makuhari Messe**

Organizer: Computer Entertainment Supplier's Association (CESA)
Co-organizer: Nikkei Business Publications, Inc. (Nikkei BP)
Ministry of Economy, Trade and Industry (International business section)

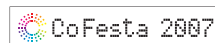
Business Day

Open to the Public

tgs.cesa.or.jp/english



Nikkei Business Publications, Inc.





MICK WEST

» THE INNER PRODUCT

DOMAIN-SPECIFIC LANGUAGES

A DOMAIN-SPECIFIC LANGUAGE

(DSL, also called a “little language”) is a language that’s intended for a very specific use. It can be a programming language such as the Turtle aspects of LOGO that defines a very limited set of actions (drawing lines), or it can be a data definition language that encapsulates the representation of some presentable data, such as graphics or sound. HTML can be thought of as a domain-specific language as it’s limited to describing the presentation of a web page.

This article looks at the potential uses of DSLs in games. I’ll look at a specific domain and create a language for it, as well as discuss some of the problems I encountered.

SPECIFIC VS. GENERAL

A DSL differs from a general-purpose language in that the latter must support a large amount of functionality, including: variables, data structures, conditional expressions, looping constructs, and functions. General-purpose programming languages may also support various forms of abstraction, object-oriented programming, lambda expressions, and so on.

DSLs can be classified as either internal or external. An internal DSL is simply an extension of an existing general-purpose language. You can think of an internal DSL as simply being a set of functions,



FIGURE 1 Rodney A. Greenblat’s “Lunar Module” is part of the domain for a new domain-specific language.

data structures, and conventions applied to an existing language, such as C++ or Ruby. This set of functionality is still specific to one problem domain. A typical internal DSL might be one used to define state transitions for AI using a set of query functions and a switch statement. Many games have implemented this kind of system in the game code in C++, as AI is often the responsibility of a programmer rather than a designer.

An external DSL is an independent language that has been entirely created for this specific purpose. Generally, a DSL program will be a text file, which is then interpreted (or perhaps compiled) by some part of the game engine or tool chain. Again, AI is a common usage of a DSL—when programmers hand off AI to a designer, they will frequently make it more data-driven, often to the extent that they supply a “little language” to script the AI transitions.

style and allowed for easy creation of similar pieces for use in-game.

The first step in creating a DSL is to get a rough idea of the elements that the domain comprises. Looking at the Elemental works, we can see a number of common aspects. There are concentric oval shapes, with petals adjoined to various sections. Many of the works have segmented circles with colored circles inside them. There are little propellers and various other shapes that repeat both within individual works and within Greenblat’s overall collection.

I decided the best way to approach creating this DSL would be to pick one piece and attempt to replicate parts of it. I chose the painting “Lunar Module” (see Figure 1). Many common elements hold the piece in its style: solid circles, concentric ovals with color gradients, petals, and stars. Being even more selective, I isolated one corner of the

THE DOMAIN

In experimenting with domain-specific languages for this article, I defined my domain as the works of Rodney Alan Greenblat, the artist responsible for the unique characters and world design in PARAPPA THE RAPPER and UM JAMMER LAMMY.

Greenblat has a large body of artwork with a very distinctive whimsical style. For my specific domain, I picked the artwork from his Elemental tour, a collection of semi-abstract paintings in a distinctive brightly colored and geometric style. The idea was this: If such a style of artwork were to be used in a video game, then it might be very useful to have a DSL that encapsulated that

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 This detail from the lower left corner of "Lunar Module" serves as the basis for the code used to replicate its style.

painting, the purple box with blue petals [see Figure 2].

WHIMSY CODE

For my first iteration, I decided on "build" rather than "extend." I wanted the creation process to be interactive so I could edit the "code" in real time and immediately see results. I was also not sure about the level of abstraction I was going to use, and I wanted the language to be very loose, unconstrained by syntax requirements. I also wanted to have more control over the speed of execution, so I decided to build a language parser using C++. I decided to call this language Whimsy and use .whimsy as a file extension to identify programs in that language [for example, lunar.whimsy].

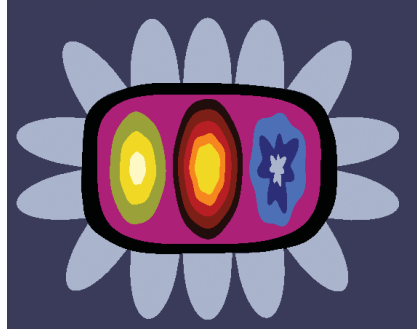


FIGURE 3 Reproducing the detail using SuperEgg, Inner and Petal primitives of the Whimsy language.

The initial exploratory coding was fairly straightforward. The code would read in a file, split it into lines, split the lines into tokens, and then simply parse the lines with a series of "if" statements; a segment of the code is shown in Listing 1. [The complete code can be downloaded from www.gdmag.com.]

At its most basic description, Greenblat's painting is composed of colored shapes. There are concentric shapes within shapes, and some shapes have other shapes attached to them. There are shapes that split other shapes, and shapes that are lists of other shapes. It seemed to me that some form of hierarchical list of shape objects was needed.

I created an abstract base class of shape called a Whim (class CWhim), and from this I derived an abstract ConvexShape object and then derived other shapes, collections of shapes, and sub-shapes from these base classes. The "world" of a painting is just a std::vector container of these objects.

After the obvious primitives of rectangles and circles, I needed a way to create the ovals. I created a CWhim called CSuperEgg. A SuperEgg is a term for the solid version of a SuperEllipse, which is a shape defined by the equation $(x/a)^r + (y/b)^r = 1$, where a and b are the length of the axes, and r defines the curvature. [See Wolfram in Resources for a full explanation.] I added a SuperEgg keyword to the parser and a little bit of code to read the parameters, and created the object.

Equations like the one above produce very nice squared ovals, but they are too precise to match the more freeform ovals in the paintings. To create a less precise shape, I created a new "distort" object, which simply takes a parent object and overrides the render function to add some periodic noise displacement from the center point. At a low frequency, it simulates the hand-drawn look of the original and can be applied to any of the ConvexShape primitives.

Next I added the petals. These simply took a parent ConvexShape object and

LISTING 1 ad hoc DSL parsing code

```

if (token == string("rectangle"))
{
    debug_log("NEW RECTANGLE");
    CWhim *p_whim = new CRectangle();
    Add(p_whim);
    if (!grouping)
        m_parse_context.clear();
    m_parse_context.push_back(p_whim);
}
if (token == string("at"))
{
    float x =
(float)atof(tokenizer.NextToken().data());
    float y =
(float)atof(tokenizer.NextToken().data());
    m_parse_context.back()-
>SetPosition(Vector2(x,y));
}
    
```

LISTING 2 whimsy code

```

superegg 0.15,0.10,3.5 at .3,.7 size 1.2 black distort
.01
petals 14 0.05 size 1.8 petalblue
inner .88,.01 tvpurple
superegg .1,.2,2 at .20,.7 size .4 distort .01 tvlime
inner .65,.01 tvyellow
inner .45,.01 tvlightyellow
superegg .1,.2,2 at .3,.7 size .5 distort .01 tvblack
inner .85 tvbrown
inner .80 tvred
inner .75 distort .03 tvorange
inner .70 tvyellow
superegg .1,.2,2 at .4,.7 size .4 distort .05 tvblue
inner .6 distort .2 tvdarkblue
inner .4 petalblue
    
```

A SuperEgg (with petals) contains other supereggs. See Figure 3 for the rendered results.



FEBRUARY 18-22
2008
SAN FRANCISCO

GDC 2008

**MOSCONE
CENTER**

www.gdconf.com

VISIT GDCONF.COM
TO CHECK OUT
2007 event photos, videos,
and session proceedings





FIGURE 4 Some examples of the flexibility of the Whimsy primitives.

positioned themselves at specified points along the perimeter. All ConvexShape objects (including the distorted shapes) have a member function that returns points at a given angular or linear distance around the perimeter, so the attachment points and base profiles of the petals can be calculated easily.

The height of the petals is specified as a multiple of the width of the base. I found that as much as possible it was best to keep all numbers relative to some other object or part of an object, as it makes dependent changes far easier.

Finally, I added an Inner shape, which simply takes one ConvexShape and creates a copy of it inside, shrunk by a certain ratio and optionally distorted. Using multiple Inner shapes made it very easy to reproduce the concentric multi-colored ovals.

These few primitives allowed me to reproduce, in part, a segment of “Lunar Module” (see Figure 3). The code used to generate the mock-up is shown in Listing 2.

LISTING 3 five whimsy creations

```
superEgg .1,.05,4 at .2,.2 green distort 0.01
petals 8 blue

superegg .1,.1,2 at .5,.2 grey distort .02
inner .5 orange distort .02
petals 12 blue size 3

circle at .35,.35 size .05 distort .04 petalblue
petals 8 size 20 petalblue

superegg .1,.05,4 at .2,.45 white distort .1
inner .5 distort .01 yellow

superegg .1,.1,3.0 at .5,.45 brown
superegg .1,.1,2.0 at .5,.45 teal
superegg .1,.1,1.0 at .5,.45 darkblue distort .02
petals 20 .5 yellow size 3
superegg .1,.1,0.5 at .5,.45 darkgreen
petals 50 orange
superegg .1,.1,0.25 at .5,.45 purple
```

See Figure 4 for the rendered results of this code.

WILL THE REAL WHIMSY PLEASE STAND UP?

This fairly short program gives a reasonable facsimile of a small segment of the painting. It's lacking the rectangular highlights in the petals, and the blue shape on the right is just a dummy, since I did not get around to triangle shapes. It's also missing other elements such as the legs and the streamer on the top right. Clearly, a lot more has to be added to the language to be able to create the rest of the painting or the others in the series.

But already we have a quite powerful set of primitives. Although each set is only a few lines of code, you can get a variety of results from fairly simple code. This kind of power is something that often does not come up until you put a tool such as this DSL in the hands of an artist and give him or her a chance to play around with the parameters and the code. (Listing 3 and Figure 4 contain more examples of Whimsy.)

One rather surprising result was that I was able to duplicate the star-like objects using a tiny distorted circle with eight very long petals. Ideally these would have been created from four intersecting brushstrokes, but the distorted petals works pretty well.

The stars raise a central issue. There are 14 stars in the painting. Clearly it's inefficient to create each one by hand by cutting and pasting the same circle and petal code. You really want a Star function that can manage the position, color, and size parameters. We need some kind of macro or function definition functionality.

Although such functionality would not be particularly hard to add, if we had started out by extending an existing language rather than creating our own unique interpreted language, then we would already have this functionality, as well as looping constructs, expression evaluation, variables, and more. I could quite easily have implemented the Whimsy programs in C++ as a series of function calls with very similar functionality. The problem is that I would have had to recompile and run the program every time I made a change. Right now, the results of a change are displayed the instant a file is modified,

which provides a far more interactive experience, and rapid iterations.

I could have implemented Whimsy as an extension to a powerful scripting language, such as Ruby or Python. This would have allowed me to still have the near-instant feedback; but the downside is the syntax is more restrictive. By implementing an original language, you can express it in any form you like. If you base it on Python, for example, you have to accept many of the restrictions of Python. Ruby is better, but it still comes with some syntactical constraints. There may also be performance implications of using a language like Ruby.

GRAPHICAL INTERFACE

One reason I like the idea of defining an original DSL is that it can be more easily extended from a text-based editing tool to a graphical-based editing tool. Listings 2 and 3, for example, show that there are a lot of numerical parameters. These are tedious to change by hand, and it's a logical extension to allow some form of direct graphical editing of things like size and position.

You can think of this technique as a graphical interface to the text. GUI tools can be used to edit the relatively high-level DSL code, and still have it human readable (and editable). Used in this way, a DSL can be a useful intermediate step in tool development.

In future articles, I'll explore implementing Whimsy as an internal DSL using Ruby, as well as extending the available primitives to cover a larger range of Greenblat's work. I'll also investigate the practical process of overlaying a GUI over a DSL. ❧

REFERENCES

Greenblat, Rodney:
www.whimsyload.com/home.html

Mathematical description of Super Ellipses:
<http://mathworld.wolfram.com/Superellipse.html>

Fowler, Martin. Domain Specific Language. February 13, 2004.
www.martinfowler.com/bliki/DomainSpecificLanguage.html

Austin Game Developers Conference



Save the Date

September 5-7, 2007

Austin Convention Center | Austin, TX

Visit AustinGDC.net to access:

- Official Conference Preview Guide
- The Results of People's Choice Voting
- Up-to-the-minute Event Announcements



From the Virtual World of Habbo
Sulka Haro
Lead Designer, Sulake Corporation



From World of Warcraft®, Diablo® and StarCraft®
Mike Morhaime
President: Blizzard Entertainment, Inc.

Join us in Austin to learn about the hottest issues facing online games, audio for games, and writing for games

EARLY REGISTRATION IS NOW OPEN

Register Now Through 8/1/07 and Save 40% on Conference Passes

www.AustinGDC.net

DAILY GAME EVENT COVERAGE



GAMASUTRA

The Art & Business of Making Games

Your Comprehensive Source For

Game Industry

INFORMATION



Timely Business & Management Updates
Breaking News From Seasoned Editors
In-depth Features by Industry Leaders
Crucial Career & Education Resources

**SIGN UP FOR NEWSLETTERS
POWERED BY GAMASUTRA!**

GAMASUTRADAILY

CAREER NEWSLETTER

INSIDEGAMASUTRA

**GAMES
ON DECK**

**GAME
CAREERGUIDE.COM**

**SERIOUS
GAMES SOURCE**

indiegames.com



CMP
United Business Media



STEVE THEODORE

PIXEL PUSHER

FIGHT THE GOOD FIGHT

Against crunch time

EXTERIOR: NIGHTTIME, A MUDDY TRENCH.

An ARTIST and an ANIMATOR are checking their watches. They are unshaven and have dark circles under their eyes. Their equipment is threadworn and dirty. The distant drumbeat of artillery fire is punctuated occasionally by the crack of a rifle bullet. A nattily dressed PRODUCER approaches with a forced smile and a clipboard under his arm.

PRODUCER: What ho, lads? Ready for the big push?

They don't respond.

PRODUCER: Don't worry boys, HQ has it all worked out. One quick push and then in stores by Christmas, eh? Everyone pulling together! Jolly good fun!

ANIMATOR: Yes sir, whatever, you say sir.

ARTIST: It's not ... not going to be like last time is it sir? I never thought I'd see home again after last time ...

PRODUCER: What rot! Higher ups have it all under control. It'll be a cakewalk ... Greeted with cheers and flowers, eh? Right ho, carry on lads. See you at the release party!

He strolls on, consulting a clipboard, humming a martial tune.

ANIMATOR: (thoughtfully) I've got a little girl at home what I ain't never seed yet.

ARTIST: You married, gov'nor? Why, I never knew ...

He's interrupted by the screech of a whistle. Starshells rise on every side. They pull their helmets low and shoulder their heavy packs of Mountain Dew, Cheezits, carpal tunnel wrappings, and bug reports. As they mount the ladder to go over the top, the ANIMATOR turns to the ARTIST and says, quietly

ANIMATOR: Do you believe in royalties, mate?

The ARTIST starts to answer, but what he says is lost as a shell goes off, showering them with priority-one bugs and unhelpful creative feedback. Stricken, the ANIMATOR falls. The camera pans across the field as muddy, overloaded game developers flounder slowly toward the barbed wire across the battered moonscape of Release Candidate One.

Fade to black.

CRUNCHY REVELATIONS

The game industry is famous for a lot of things—creativity, innovation, for having an intravenous hookup to the brains of the world's teenagers—but not all of them are positive. The now infamous EA_Spouse controversy was only remarkable because it drew the attention of the wider world to something we all know too well: the looming menace known as crunch time.

Crunching and game development are almost synonymous. I'd bet that first prehistoric version of SPACE WAR involved a bunch of bleary late night sessions fiddling with vacuum tubes and punch cards over whatever passed for Diet Coke in 1957.

The IGDA has been banging the drum about crunch time and its impact on our lives and careers for many years (the

Quality of Life section of the IGDA web site has some interesting and sobering reading for the curious). GDC talks on developing without crunches crop up as regularly as daisies. And yet, somehow, crunch time remains an institution, a ritual, a rite of passage, and a tremendous pain in the neck.

There are a number of schools of thought about why crunch remains such a fixture of our industry. Economic determinists will point out that we don't—or at least we didn't, until the EA lawsuit—get paid more for working more hours. Anything without a price tag attached will be used wastefully, including our nights and weekends, my brothers and sisters.

Some folks argue that Hollywood, despite unions, overtime, and work rules galore, is almost as given to insane deadlines and life-sucking work maelstroms as we are. They say it's the nature of a creative business.

Our spouses and significant others might suggest that the real problem isn't the industry at all—it's us. We're a bunch of workaholics that hide in our offices to avoid real life.

A lot of us will blame our managers or publishers for being too optimistic, overeager, or just plain amateurish. As befits a big problem, there seems to be plenty of blame to go around.

Whether you regard the persistence of crunches as evidence that the industry still has a lot to learn about professional management practices, or as the by-product of working in a frantically evolving medium, or as an evil plot by sinister capitalists, the result is pretty much the same. At some time in the next 24 to 36 months, you are almost certain to be crunching. Whoever you blame, it's dollars to doughnuts you're going to have to live through at least one more crunch. How do we artists manage to stay alive and

STEVE THEODORE has been pushing pixels for more than a dozen years. His credits include MECH COMMANDER, HALF-LIFE, TEAM FORTRESS, and COUNTER-STRIKE. He's been a modeler, animator, and technical artist, as well as a frequent speaker at industry conferences. He's currently content-side technical director at Bungie Studios. Email him at stheodore@gdmag.com.

sane through yet another industry death march?

ONE PLACE AT A TIME

Crunch time locks us away from our families and social lives. That physical separation, though, is only part of the cost. The obsessive compulsive streak that makes us artists often keeps us busy planning and scheming about our work even when we're nominally off on our own. In ordinary times this isn't usually a healthy habit, as you can probably learn pretty quickly by asking a spouse, significant other, or family member. When crunch time rolls around, though, that touch of absentmindedness can quickly turn toxic.

Psychologists have consistently found that worrying about problems you are physically unable to address is much more stressful than actually working on them. Physiologically, lying awake in bed at night thinking about how you're going to redo the vertex assignments on Sammy the Sheep tomorrow is more stressful than actually slogging it out in the Skin Modifier dialog box.

The upshot is that when crunch time comes around, any precious scraps of free time that do come your way must be guarded ferociously. If your boss tells you to go home and see your family (something good managers do surprisingly often), you need to discipline yourself to leave work at the office for the duration. If not, you're just rendering yourself, and most likely your family as well, even more freaked out and irritable than you would be back at your desk. As Buckaroo Bonzai once said, "Wherever you go ... there you are."

THE BIG PICTURE

Dissecting that fresh-from-GameStop copy of a competitor's game on the office plasma TV is (as crunching is) one of the basic rituals of the games business. The ability to quickly intuit the essence of a game is what separates the pros from the fanboys.

Sadly, though, our genius for understanding other people's games doesn't always extend to our own projects—we may know the characters, the settings, and all sorts of arcane plot

details, but it's easy to get lost in our own little projects and lose sight of what any fan would know instantly about atmosphere and style. Parochialism is always a problem, but when crunch time looms, it turns serious.

Crunch time is when all the marvelous possibilities of a project get ground down into the realities of a shipped product. More decisions are made in the last six weeks of a project than in the first six months, and in the usual atmosphere of perpetual crisis, the odds are good that you're going to have to make some of those decisions yourself, most likely on short notice and with little or no support from above. No matter how humble your job title, crunch is the time when you're likely to be told, "Just fix it," and left on your own.

Without a strong sense of what the product needs and wants, you'll have a hard time making the right call. You should be actively trying to learn the tao of your project long before the crunch hits, so you'll be ready to operate on your own when headquarters is under siege. Having a sense of your stylistic and technical duties is also a strong defense against crunch stress. 2:08 a.m. five days before RC1 is no time to be groping for a personal vision.

THE BIGGER PICTURE

Knowing the larger picture isn't just about channeling your art director or your lead designer's vision. Art teachers have yammered on about the starting with big gestures and working down to details since the first Cro-Magnon pointed out to a Neanderthal that bison usually have four legs.

Drilling down into details too early, before the basics of a painting, model, or animation are well established is tempting. When the crunch monster is upon you, though, sticking with the big picture becomes vital. You can't be ruthless enough to keep up with deadlines if you're over-invested in little details. If you can't let go of the beautifully rendered rain stains on a concrete wall texture, you'll never get the rest of the building done, and all that work will go for naught.

This advice may sound trite, but it's still hard for a lot of artists to swallow. The grim

gothic romance of crunch time thrives on anonymous late night heroics and last minute touches of fanatical love. Many artists fear to put a brake on their obsessions lest they lose the special touch that elevates mere product into something like art. If none of us ever stayed at work until 3 o'clock in the morning to satisfy our creative itches, it's true our games would be a lot less compelling and beautiful. But if we all were a bit more deliberate about our priorities, we might be doing some of those grace notes at 11:30 instead of 3:00, and still get some sleep.

PACE YOURSELF

Which brings us down from the realm of art theory to a more earthly problem: stamina. Crunch is a physical and mental ordeal. Some of us stay late for fear of the boss, some of us do it out of loyalty to our teammates, a few of us do it out of misplaced machismo. But all of us suffer from it.

If you don't think crunching affects the quality of your work, go back and look through your source depot for the detritus of the last big push. It's probably full of weird little typos, files with unprintably profane names, and strange lapses in logic, not to mention buggy code and inadequate art.

When you push your body beyond reasonable limits, you act slower and stupider. Economists call it "the law of diminishing returns," and we call it "burnout." But the results are the same. No secret reservoir of mental toughness or dedication to the game will make this fact go away. The only defenses are a decent ration of sleep, some exercise, and a bit of life outside the office.

As crunch time builds to a crescendo, watch yourself carefully for signs of exhaustion. If you're starting to make mistakes and stumble, throttle back. Are you staring at a file folder and don't remember why you opened it? Did waiting 30 seconds for a render turn into a five-minute zone out? Then go home!

There's a lot of good information on the IGDA's web site that can help you set reasonable goals. Some of the research might also come in handy when talking to overeager managers about your schedule. Learn to spot when you've hit

your point of diminishing returns and pack up for the night. Over the course of a multi-week crunch you'll easily come out ahead by any measure of productivity you or your bosses care to name.

LEARNING

Measuring productivity has a nasty, un-artistic sound to it, but it's actually a valuable weapon for fighting back against the crunch monster. One of the most important ways you can protect yourself is to get a good grip on your own scheduling, starting with keeping a spreadsheet of what you did and how long it took. This isn't for management, this is for you. These records will give you a better chance of spotting impossible workloads a long way off and making sure they don't get scheduled as a matter of course. A good journal won't make your current crunch go away, but better

estimates might make the next one a little less grueling. Those who cannot learn from history, it is said, are doomed to repeat it. If the industry as a whole hasn't done a very good job of learning to cope with crunch time, that doesn't mean you can't do better on your own.

FIRST THINGS LAST

It would be great if the industry came to its senses and abolished crunching. Hopefully, some combination of better management, tougher negotiating by developers, and the fallout of the recent lawsuits will consign Red Bull poisoning and the 5 a.m. check-in to the dustbin of history alongside child labor and the Sega Dreamcast.

Until that day, we have to look out for ourselves. Even more vitally, we have to look out for our families and loved ones who are usually the real victims of the

institution. We, at least, are rewarded for our sacrifices with (some) money, (slightly) better jobs, and the (grudging) respect of our peers. Our families get nothing beyond the occasional company picnic. All the cheerful careerist cynicism in the world can't compensate for the realities of broken relationships or neglected kids. There's never been a game that was worth a single divorce, but anybody with a few shipped titles on their resume could name you a title or two that cost that much.

That's why it's up to us to grapple the crunch monster. Growing out of our frat-house work habits isn't just good advice—it's a deadly serious duty. The family wants you home from the trenches, safe and sound, not just another anonymous game industry hero in the small print on Mobygames. Don't disappoint them. ❖

Game Developers Conference China

GDC CHINA
August 27-29, 2007 • Shanghai

Presented By
CMP (United Business Media) and IDG (INTERNATIONAL DATA GROUP)

GDC China offers an insider's perspective on key regional issues such as outsourcing, next-gen game design, and mobile, online, and casual gaming. It is essential for both Asian and Western companies who want to drill down into game development in the Chinese market.

GDC China also provides an unparalleled opportunity for networking with key figures in both Chinese and Western markets.

REGISTER TODAY!

www.gdcchina.cn/en

DIAMOND SPONSORS:



PLATINUM SPONSOR:



GOLD SPONSORS:



WELCOME BANQUET SPONSOR:





GET YOUR GAME ON

EVERYTHING YOU NEED TO KNOW TO GET INTO THE GAME INDUSTRY!

- News and Features for students and educators
- Directory of Game Schools
- Getting Started section – an invaluable how-to guide
- Message Boards
- Faculty & Student Research – Salary Survey

You can purchase your digital edition of the 2007
gamedeveloper Game Career Guide online at

www.gamecareerguide.com





GAME SHUI

HYBRID VIGOR

THERE IS A PRINCIPLE OF GENETICS AND selective breeding of plants and animals known as heterosis, or more commonly, hybrid vigor. Simply stated, the breeding of two distantly related parents can result in an offspring that combines the best qualities of both. It is in some ways the opposite effect of inbreeding, where two closely related parents can yield offspring with genetic defects.

I was reminded of this principle while playing PUZZLE QUEST: CHALLENGE OF THE WARLORDS, designed by Steve Fawkner who was the designer of the popular WARLORDS series of games from SSG. PUZZLE QUEST is a new style of game that combines two very familiar, but on the surface, totally unrelated gameplay genres—a basic fantasy role-playing game (FRPG) with the requisite brave warriors, knights and mages facing menacing orcs, trolls, and dragons, and a match-three game, similar to BEJEWELED. To be blunt, both of these components individually are familiar to the point of boredom—and yet the way they combine and reinforce each other is startlingly fresh and effective in PUZZLE QUEST.

A TALE OF TWO GAME GENRES

The design principle that makes PUZZLE QUEST work originates in the synergy between the two genres. The FRPG backdrop provides a meaningful setting and context for the match-three play, and the match-three element provides a fun core gameplay mechanic that is used in five simple variations: combat (the primary game element), capturing opponents, learning spells from captured enemies, training mounts, and creating magic items.

These two well-known gameplay conventions are not too exciting separately, but it is surprising just how well they work together. The match-three gameplay is quite versatile—the different variations mentioned above all build on common skills, but are each unique in interesting ways. They provide a fresh, if abstract way to accomplish these standard FRPG tasks, bringing a sense of discovery to that genre.

Conversely, the FRPG conventions provide something very important to match-three gameplay: context. Like other FRPGs you can use magical Mana of different colors to cast different spells. But in this game you get that Mana through puzzle play.

There are seven basic symbols to match, like the gems in BEJEWELED: red, green, blue, or yellow mana; purple stars (experience points); gold coins (money); and skulls (damage in combat). In BEJEWELED, I don't particularly care which set of gems I match next, but in PUZZLE QUEST I may need more red mana for a particular spell, or I may wish to pick up the available blue mana to prevent an enemy troll from using it in a regeneration spell, or I may focus on skulls to attack my foe.

There's more to this synergy than I have space to cover here, so I'll simply note that it's a fascinating principle that has a lot of potential. The blending of different genres has been used before to good effect. The real-time strategy genre

can be seen as a combination of the classic turn-based wargame, and the real-time building gameplay of the original SIM CITY. What hybrid vigor from genre-combinations might we see in the future?

BREEDS TO COME

In my May 2005 column I talked about how the upcoming game SPORE creates synergy among three interlocking gameplay mechanisms: user-developed content, multiple gameplay genres, and a gameplay mechanism inspired by evolution of procedurally defined game assets. It is easy to see how PUZZLE QUEST's example of blending a well-established game genre with a casual game mechanism could spawn new types of games.

Consider a wargame and match-three blend where, instead of colors of mana, one matches symbols to generate fuel, types of ammunition, and weapons, or perhaps components of combined arms attacks. Or one might use an adventure game to provide an interesting story and characters, but instead of the often frustrating puzzles built into the storyline, the gameplay can advance by solving match-three games that help one defeat enemies, find critical objects or tools, or unlock doors. Or consider combining the RPG style with a different casual game dynamic, like the customer service in DINER DASH to create a science fiction RPG where you play a jack-of-all-trades working your way across the galaxy by serving as crew on interstellar "tramp steamers," repairing the engines, aiding the passengers, and even serving dinner to passengers along the way. Or perhaps a racing car/DINER DASH combination where you drive the cars, but also control the pit crews.

Some of these combinations may not succeed, but others could demonstrate hybrid vigor and revive flagging game genres, possibly even supplanting them. It's a great opportunity for designers hoping to create new gameplay from old formats. ❖

Puzzle Quest's genre-bending is lampooned in this Penny Arcade comic.

(www.penny-arcade.com)



NOAH FALSTEIN has been a professional game developer since 1980. His web site, www.theinspiarcy.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.



JESSE HARLIN

» AURAL FIXATION

VOX POPULI

What players think of game audio

LAST YEAR I SPOKE TO GAME REVIEWERS

regarding their thoughts and opinions on game audio ["Developers, Meet your Reviewers," June/July 2006]. This year I decided to broaden the scope and hear from the gaming public at large. They're the consumers of our products, and the engine that drives word-of-mouth sales. I wanted to know how gamers—both casual and hardcore—perceive game sound.

I created a short online poll which ranked the importance of sound effects, music, and voice as parts of the consumer's overall gaming experience. Then, in order to cast the widest net possible, I enlisted a small army to disperse the poll through MySpace, Facebook, and the blogosphere (special thanks to Kotaku.com).

Here's the fine print—I'm not a statistician and I make no claims that this is the most scientific polling ever conducted. It should also be noted that specific market research conducted by large companies such as Microsoft, Sony, or EA may be completely different from the results my poll collected. However, with nearly 2,000 responses, these results should offer an interesting cross-section of how the public regards game audio.

CRASH

Of the three audio categories, sound effects are the most well-received. 63 percent of respondents had never turned the effects off while playing a game. Additionally, 87 percent of respondents felt as though they had played a game that could not possibly be completed without sound effects. Unfortunately, this means that 13 percent of the respondents have never played a game where they've

viewed sound effects as an essential part of their gaming experience.

In the comment threads that sprung up in response to the poll, effects were by far the most highly-praised of the three audio disciplines. First-person shooter fans often attributed their elite skills in large part to audio design. One MADDEN fan noted an inability to play defense without audio turned on. Not surprisingly, sound effects were cited as contributing to the immersive atmosphere of games like SILENT HILL, BROTHERS IN ARMS, and BURNOUT REVENGE.

Praise for the transformative experience of a decent surround sound system was the most overwhelming. Interestingly, 5.1 systems were repeatedly mentioned as being more important to next-gen gaming than a high-definition television screen. Those who left comments that most enthusiastically praised sound as a fundamental part of gaming were also frequently those touting 5.1 as "the new standard" for game audio.

POP

Music proved to be significantly more subjective. The numbers were exactly flipped from those of sound effects when it came to tuning out—63 percent said that they had turned off the music in the past. Surprisingly, even with the success of games like GUITAR HERO and DANCE DANCE REVOLUTION, or the inclusion of music-based puzzles in best-selling titles such as GOD OF WAR or THE LEGEND OF ZELDA: THE WIND WAKER, only 56 percent of the respondents felt as though they had ever played a game where the music was absolutely necessary for its completion.

Like sound effects, music was commonly cited as a major contributor to a game's mood. A "good soundtrack" was often credited as being a leading factor toward immersive gameplay. Licensed music, however, was a frequent target of criticism. Most respondents who expressed a desire to hear a song-based

soundtrack as opposed to a game-specific score mentioned that they simply turned the game's music off in favor of their own external playlist.

That said, for Xbox and Xbox 360 users, only 56 percent of those responding said that they had used the custom soundtrack option to add their own music into their games. At the same time, 58 percent said that they wish more games offered the option of adding their own soundtracks. This was also a common topic in the comment threads. Rather than poorly composed or produced music, the biggest complaint was a simple lack of enough music to cover the ever-expanding scope of today's games.

BANG!

Admittedly, voice was the smallest portion of the poll. As dialogue can rarely be turned off separately from the sound or music, my main concern was with the public's perception of an often-maligned aspect of audio by game reviewers: repetitive dialogue.

Happily, when asked whether they find game dialogue to be engaging, too repetitive, or downright obnoxious, 72 percent of the respondents felt that voice acting is either "entertaining" or "worth listening to." More encouragingly, 70 percent expressed a wish that more games contain recorded dialogue.

IN THE END

What is evident from all these responses is that gamers recognize and largely appreciate the strides being made across the industry to continually raise the quality of in-game audio. Regardless of the successes, however, the realm of game audio remains one in which users are commonly altering the originally intended listening experience, either by changing our mixes via in-game volume sliders or replacing our soundtracks all together. Whether they like everything they hear or not, it's clear from these 2,000 responses that THX is correct: The audience is listening. ❖

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.

BLACKSITE
AREA 51

JOIN US

NOW RECRUITING:

Programmers:
Gameplay, AI, and Engine

Sr. Programmers: all areas

Lead Designers

Motion Graphics/UI Artists

Sr. and Lead Character Artists

Sr. Environment Artists

Sr. and Lead Animators



**THE FUTURE
OF VIDEOGAMES**

LEARN MORE AT JOBS.MIDWAY.COM

Visit our booths at Siggraph and the Austin Gaming Conference.



Visit www.esrb.org
for updated rating
information.

Blacksite Area 51 © 2007 Midway Games West Inc. All rights reserved. BLACKSITE AREA 51, and AREA 51 are trademarks or registered trademarks of Midway Games West Inc. Used by permission. MIDWAY and the Midway logo are trademarks or registered trademarks of Midway Amusement Games, LLC. Used by permission. Midway Home Entertainment Inc. and its affiliates do not monitor, endorse or accept responsibility for the content of any non-Midway website. Distributed under license by Midway Home Entertainment Inc. This notice contains a "forward-looking statement" within the meaning of the Private Securities Litigation Reform Act of 1995 concerning future business conditions and the outlook for Midway Games Inc. (the "Company") based on currently available information that involves risks and uncertainties. The Company's actual results could differ materially from those anticipated in the forward-looking statement as a result of these risks and uncertainties, including, without limitation, the Company's ability to maintain the scheduling of product introductions as more fully described under "Item 1 - Business - Risk Factors" in the Company's Annual Report on Form 10-K for the year ended December 31, 2005, and in any more recent filings made by the Company with the Securities and Exchange Commission. The forward-looking statement speaks only as of the date on which it is made, and the Company undertakes no obligation to update it to reflect events or circumstances after the date on which it is made or to reflect the occurrence of anticipated or unanticipated events or circumstances.

**Southern California,
U.S.A**

NEVERSOFT

**TONY
HANK'S**

**GUITAR
HERO**

Looking for;

Programmers

Artists

Designers

Tech. Artists

UI programmers and more...

www. NEVERSOFT .com



ROCKSTAR GAMES IS HIRING

- PROGRAMMERS
 - ARTISTS
 - ANIMATORS
 - DESIGNERS

IN THE US, UK, AND CANADA
FOR OUR HIGH PROFILE
NEXT GENERATION TITLES

WWW.ROCKSTARGAMES.COM/JOBS

NYC • TORONTO • SAN DIEGO • EDINBURGH • VANCOVER • LINCOLN • LEEDS • LONDON

RESUMES@ROCKSTARGAMES.COM
WWW.ROCKSTARGAMES.COM



Join the PlayStation® team!

Be a part of the most exciting and innovating computer entertainment in North America. Sony Computer Entertainment America (SCEA) markets the PlayStation® family of products and develops, publishes, markets, and distributes software for the PlayStation®3 console, the PlayStation®2 computer entertainment system and the PSP® (PlayStation Portable).

We are currently looking for talented and experienced game development professionals for the following areas:

Administrative Support
Artists & Designers
Art & Production
Game Programmers
Marketing
Merchandising

PlayStation® Network
Product Marketing
Promotions
Quality Assurance
Software Engineers
Tools & Technology

At SCEA, we offer our employees exceptional benefits, an open and creative working environment, and the opportunity to be part of a stellar team developing high profile action First Party titles. For additional information about our exciting opportunities, please visit our website at:

www.us.playstation.com/jobs

With opportunities in these areas:

Foster City, California

Santa Monica, California

Salt Lake City, Utah

San Diego, California

Bend, Oregon



SONY



COMPUTER ENTERTAINMENT®

PLAYSTATION® 3

PlayStation® 2

PSP®
PlayStation Portable



"PlayStation" and the "PS" Family logo are registered trademarks of Sony Computer Entertainment Inc. Equal Opportunity Employer.
The Sony Computer Entertainment logo is a registered trademark of Sony Corporation.



GET ON BOARD!

We have immediate openings for:

Animators
Art Directors
Character Modelers
Designers
Environment Artists
Graphics Programmers
Lighters
Network Programmers
Producers
Scripters
Visual Effects Artists

Come visit us at Siggraph booth # 1737.

ACTIVISION
VALUE PUBLISHING, INC.

NEVERSOFT

Infinity Ward

Vicarious Visions

TREYARCH



Luxoflux



BEENOX

SHABA

redoctane

ACTIVISION
FOSTER CITY

Toys For BOB

WWW.ACTIVISION.COM (GO TO "CAREERS")

Work for a studio with great character



Emeryville, CA Boston, MA Newport Beach, CA Kirkland, WA New York, NY
 Vancouver, BC Austin, TX Charlottetown, PEI Eugene, OR Los Angeles, CA

Go to www.f9e.com for current job listings



Serious Games. Quality Life.

NOW HIRING SOFTWARE ENGINEERS

jobs@totimm.com

Austin, TX ★ Alameda, CA

www.totimm.com



THE FIRST NATIONAL PARK
WITH NEXT-GEN TECHNOLOGY
NOW RECRUITING ACROSS ALL DISCIPLINES

WWW.LUCASARTS.COM/JOBS



LUCASARTS™

The
centre
of
digital
media
is
here

Announcing the Masters of Digital Media Program (MDM), a ground-breaking new professional Graduate Degree in Vancouver, Canada

The MDM is an innovative, full time professional graduate degree program at the Great Northern Way Campus now accepting applications for September 2007. Graduates receive a master's degree bearing the seals of Vancouver's 4 major post secondary institutions:

- The University of British Columbia,
- Simon Fraser University,
- Emily Carr Institute of Art + Design,
- British Columbia Institute of Technology.

Students develop the professional skills required to be effective creators, practitioners, and senior managers in this rapidly growing industry.

Get full program details at
www.gnwc.ca/mdm



SIMON FRASER
UNIVERSITY



MDM curriculum has been approved by the Academic Committee of the GNWC and is in the approval processes of the Senates or Educational Councils at our four partner academic institutions: University of British Columbia, Simon Fraser University, British Columbia Institute of Technology and Emily Carr Institute of Art + Design.

PC GAMEON

SEPTEMBER 8-9, 2007
austin convention center

FIRST-EVER PC GAMES FESTIVAL

PRE-RELEASE GAMES

TOURNAMENTS & CONTESTS

GAME DEV GURUS & PRO-GAMERS

+ REGISTER EARLY
for a chance to
win cool prizes

www.pcgameon.com

SPONSORED BY:



FOUNDING PARTNERS:



PRODUCED BY:

GameDevelopers
Conference



We're looking for you.



If you're an individual with technical and artistic vision, we've got a place for you.



THQ, THQ Wireless, Rainbow Studios, Cranky Pants Games, Heavy Iron Studios, Paradigm Entertainment, Relic Entertainment, Kaos Studios, Volition Inc., Blue Tongue, Juice Games, Vigil Games, Locomotive Games, ValuSoft, Helix, THQ Studio Australia, XDG, Concrete Games, Incinerator Studios, and their respective logos are trademarks and/or registered trademarks of THQ Inc. and/or their respective owners. All rights reserved.



www.thq.com



OUR MASTER'S DEGREE HAS ONE THING ON ITS MIND.

YOUR NEXT JOB. At FIEA everything from our industry-based curriculum to our new MOCAP studio is geared toward teaching you what you need to know to become a successful video game producer, programmer or artist.

Learn the production process in our immersive, project-based curriculum and be mentored by faculty from EA™, Disney®, Microsoft® and Take Two™. And best of all? Earn a fully accredited Master's degree in just 16 months. To learn more, go to www.fiea.ucf.edu or call 407-823-2121.



Florida Interactive Entertainment Academy



MAKE MORE ENEMIES

Game Design at Vancouver Film School

The Leader.

Dave Warfield, 15-year veteran game designer for over two dozen titles.

The Know-How.

In just one year, you'll learn every aspect of game design. Your portfolio project is a playable, industry-ready video game.

The Results.

Our graduates work at top game companies including Backbone Entertainment, BioWare, EA Black Box, Next Level Games, and Propaganda Games.

vfs.com/enemies



VFS™

VFS Student work by Jason Martin



create character.

Explore a career
in one of the most
creative industries
in the world.
Stop dreaming it—
start living it.



ex'pression
COLLEGE FOR DIGITAL ARTS



Ex'pression College offers Bachelor's degree programs in:

- Animation & Visual Effects
- Game Art and Design
- Motion Graphic Design
- Sound Arts



Call **877-833-8800** or visit
www.expressioninfo.com

You may have the drive to make games, but developing the knowledge and ability to achieve the professional standards demanded by employers is an extremely challenging path.



Building
on your **Passion**

DigiPen brings it all Together

At **DigiPen Institute of Technology**, we believe that there are no shortcuts to a serious career in the field of digital interactive entertainment. DigiPen has combined a comprehensive curriculum and world-class faculty to provide a rigorous educational experience in the following programs.

Computer Science

■ BS in Real-Time Interactive Simulation

The Real-Time Interactive Simulation (RTIS) undergraduate degree focuses on the technology and computer science behind video game development, including the development of game engines, graphics, physics, artificial intelligence, and networking.

■ MS in Computer Science

At the graduate level, students' theses focus on areas such as graphics, physics, artificial intelligence, and game design.

Production Art

■ BFA in Production Animation

Extensive traditional art and animation skills are taught alongside cutting-edge industry supertools. This approach allows graduates to work in virtually any animation environment.

Computer Engineering

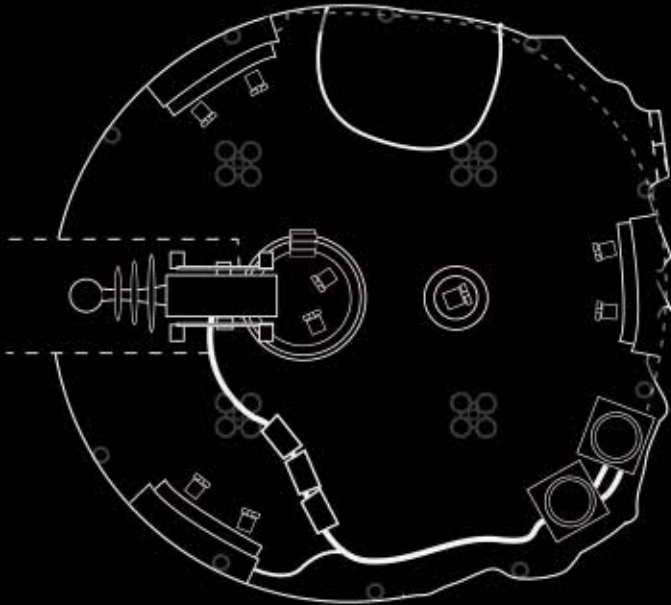
■ BS in Computer Engineering

This multidisciplinary program integrates the fields of electrical engineering and computer science with a specialized focus on video game applications, such as developing a handheld game console.



CONTACT DIGIPEN ADMISSIONS

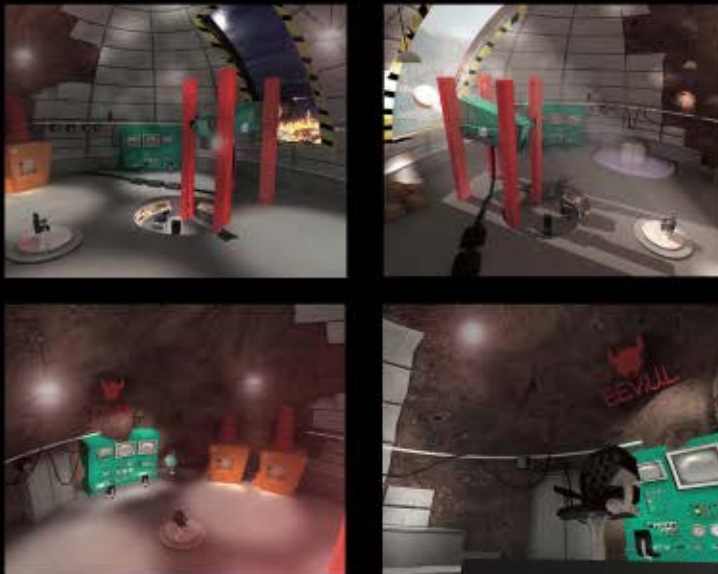
CALL (866) 478-5236 EMAIL admissions@digipen.edu VISIT WWW.DIGIPEN.EDU



E.E.V.UL HEADQUARTERS BLUEPRINTS

Got Plans?

You've got things to do. Bring your plans to life with the **Digital Arts Technology Training Institute**. Learn Video Game Art & Illustration, Character Animation, and 3D Modeling in the comfort of your own home.



Blueprints, models, textures and renders by DArTT graduate, Alexander Eigner

Prepare for your Tomorrow - Today!
Take control of *your destiny* with learning designed for *YOU*.

The Digital Arts Technology Training Institute offers *quality and credibility*, job find assistance, and 24-hour student support while you learn from home.

DArTT Offers Career Training & Diploma Programs in:

- 3D Animation
- 3D Character Animation
- Web Page Design
- Video Game Art & Illustration
- Graphic Design

Call Toll-Free **1.866.567.3010**
or visit **mydigitalcareer.com**

SERIOUS

about living your creative

PASSION?



LEARN FROM THE LEADER IN ONLINE CREATIVE ARTS EDUCATION

WWW.AIONLINE.EDU

CALL NOW: 1-877-872-8869

The Art Institute Online offers the following programs:

Bachelor: Advertising | Fashion & Retail Management
Game Art & Design | Graphic Design | Interior Design
Interactive Media Design | Media Arts & Animation

Associate: Graphic Design | Interactive Media Design

Diploma: Digital Design | Residential Planning | Web Design

ARE YOU SEEING RED?

Come to Baton Rouge and we'll get you in the Black

Baton Rouge is the largest city in Louisiana, and the only one in the nation to have a full-time office dedicated to building the digital media sector.

Louisiana can help your entertainment or digital media enterprise save between 15% and 40% on your production and infrastructure expenses.

Baton Rouge hosts LSU, the Flagship University of the state, and has a regular enrollment of 30,000. There are several other colleges and universities within 45 minutes.

Baton Rouge offers a high quality of life with exceptional education and affordable housing for your employees.



BRADIC

Baton Rouge Area
Digital Industries Consortium

225-389-7182/225-279-2540
ssimmons@cct.lsu.edu

**Come visit us in the CCT booth,
number #741 at SIGGRAPH .
You could win an iPod!**



level 14 troll

level 23 dwarf

level 31 gnome

See it in game.

Join Columbia's community of intensely creative minds.

GAME.COLUM.EDU

create...
change

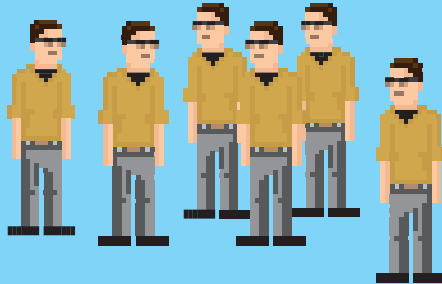


Columbia 
COLLEGE CHICAGO

INNOVATION IN THE VISUAL, PERFORMING, MEDIA,
AND COMMUNICATION ARTS



(geek)



(clustergeeking)

Please geek responsibly.

You may speak the language,
but are you geeked?
Here's a chance to prove it.

LEARN:

DIGITAL ANIMATION
DIGITAL ART AND DESIGN
DIGITAL VIDEO
GAME DESIGN
ARTIFICIAL LIFE PROGRAMMING
COMPUTER FORENSICS

GAME PROGRAMMING
NETWORK ENGINEERING
NETWORK SECURITY
SOFTWARE ENGINEERING
WEB ARCHITECTURE
ROBOTICS



www.uat.edu > 877.UAT.GEEK
877.828.4335



SCHOOL OF GAME DEVELOPMENT

COMPUTER ANIMATION > DIGITAL ARTS & DESIGN > ENTERTAINMENT BUSINESS > FILM > RECORDING ARTS > SHOW PRODUCTION & TOURING

Financial aid available to those who qualify • Career development assistance • Accredited School, ACCSCT

ONE OF THE TOP FIVE GAME-DEGREE PROGRAMS IN THE WORLD

— Electronic Gaming Monthly



800.226.7625
3300 University Boulevard
Winter Park, FL 32792

fullsail.com

© 2006 Full Sail Inc. All rights reserved.

CENTER FOR DIGITAL IMAGING ARTS AT BOSTON UNIVERSITY

WALTHAM, MA | WASHINGTON, DC

CREATE your WORLD



CERTIFICATE PROGRAMS:
3D ANIMATION | VISUAL EFFECTS | GAME DESIGN

Intensive full- and part-time programs for the skills and tools you need to turn your ideas into reality. Financial assistance and career services available. **Apply today!**

CALL 800-808-CDIA EMAIL INFO@CDIABU.COM WEB WWW.CDIABU.COM

ADVERTISER INDEX

Academy of Art University.....	63
Activation	49
The Art Institute Online	59
Autodesk	C2
Center for Digital Imaging at Boston University	62
Centre for Distance Education	58
CNET	31
The Collective Inc.	50
Columbia College Chicago	60
DigiPen	57
Epic Games	15
Ex'pression College.....	56
Fork Particle	25
Full Sail Real World Education.....	62
Havok	21
Louisiana State University	59
LucasArts	51
Masters of Digital Media.....	52
Midway Games	45
NaturalPoint	13
Neversoft Entertainment	46
Perforce Software	16
RAD Game Tools	C4
Replay Solutions LLC.....	6
Rockstar Games.....	47
Scaleform Corp.	11
Seapine Software Inc.....	3
Sony Computer Entertainment	48
Tokyo Game Show	32
THQ Inc.	53
Total Immersion Software Inc.....	50
University of Advanced	61
University of Central Fla. (UCF)	54
Vancouver Film School.....	55
Vicious Cycle Software.....	C3
Xoreax Software Ltd.....	25

**ANIMATION
AND VISUAL EFFECTS,
ILLUSTRATION, GRAPHIC
DESIGN, COMPUTER ARTS NEW
MEDIA, FINE ART, PHOTOGRAPHY,
ARCHITECTURE, ADVERTISING,
INDUSTRIAL DESIGN, INTERIOR
ARCHITECTURE AND DESIGN, MOTION
PICTURES & TELEVISION, FASHION,
DIGITAL ARTS & COMMUNICATIONS**

Image by Eva Kelenin, BEA, Photography

The best time to **THINK** about education was years ago.
The second best time is right now.

IT'S NEVER TOO LATE

REGISTER NOW

1.800.544.ARTS / ACADEMYART.EDU



ACADEMY of ART UNIVERSITY

FOUNDED IN SAN FRANCISCO 1929 BY ARTISTS FOR ARTISTS

79 New Montgomery Street, San Francisco, CA 94105

Accredited member ACICS, NASAD, Council for Interior Design Accreditation
(formerly FIDER) (BEA-IAD), NAAB (M-ARCH)

ONLINE DEGREES & CLASSES



NICHOLAS BELIAEFF

» BUSINESS LEVEL

MORE OF WHAT THEY WANT

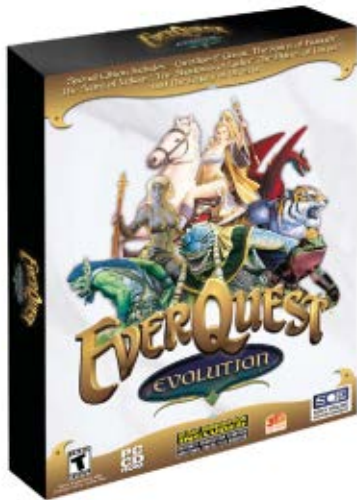
How to keep your online players coming back

MY FRIENDS AND I ALWAYS JOKE THAT

working in the game industry is like being a dog. Every calendar year you spend in games is equivalent to seven years in terms of technological developments,

gameplay innovation, stress and gray hair, and life in general.

Seeing how much the online gaming landscape has changed since we launched EVERQUEST on March 16, 1999, as one of the first 3D MMOs, you might wonder how it keeps going after eight long years (or more than half a century in dog or game industry years), with a strong player base and community.



EVERQUEST has issued several expansion packs that meet the changing needs of the players.

STRATEGY

In developing an MMO, a studio goes through all the same stages of product development that it would for a typical PC or console product, but once it ships, the team begins to deal with issues that are unique to MMOs—acquisition and retention.

In the early years of EVERQUEST, we were very focused on acquisition. How can we grow our subscriber base and continually add new people to the game?

In the recent years, our mindset switched to retention. How do we keep the players currently in the game coming back for more? And corollary to that, how do we win back players who used to play EVERQUEST? This fundamental shift changes development focus from things like new player experiences and newbie tutorials to high-end raid content, increased level caps, and alternate advancement systems.

KNOW THY GAME

Most people on the current team played EVERQUEST before they were hired. Some of our best hires have been people we knew were passionate about the game, whom we met through playing with them, on message boards, or in person. This creates a real connection between the development team and the players. This shared experience allows us to prioritize gameplay issues very effectively.

ROLL WITH THE PUNCHES

Some of the most important attributes of a successful live team are flexibility, the willingness to embrace change, and the patience to iterate over and over. People and their expectations change over time, and EVERQUEST players are no exception. EVERQUEST's audience has shifted almost an entire generation since its launch. Most of our players started playing when they were in their 20s or 30s. Now they're in their 30s and 40s. When they were in college, playing for 10 hours a day was no big deal. Now, most players have jobs, families, and other activities that make a 10 hour per day commitment completely untenable. EVERQUEST has changed over time to accommodate the fundamental changes in our players' lifestyle.

COMMUNITY OUTREACH

The EVERQUEST team has a lot of different ways to connect with the players. There are events like Fan Faire where we meet with our most dedicated players in mini

festivals that rotate around the country. We also have Dev Summits, where we invite 30 to 40 key players of EVERQUEST to interact with more than 10 people from the development team. This allows us to delve deep into issues by having multiple hour conversation brainstorming in real time on how to make the game better.

We also have our boards, email, IM, and even the phone. People on the team are in contact with players on a daily basis at the most personal level.

PERSPECTIVE IS KEY

When you have a deep connection to your players like the EVERQUEST team does, you really want to make sure they're happy. Players have some of the best ideas for items, spells, and quests, and are not shy about sharing them. There are six digits of them versus just under 40 people on the development team. The law of averages dictates that they are going to have some great ideas. The big lesson learned here is not to make empty promises or over-commit on what you can actually do once you accept that suggestion.

WHAT PEOPLE WANT

EVERQUEST had been doing two expansions every year since 2003. As our players' play patterns have changed, and those 10 hour per day play sessions gave way to shorter ones, the amount of content we released in each expansion became more and more daunting. The community asked us to slow down and adopt a "less is more" strategy. To that end, and starting with SECRETS OF FAYDWER due out November 2007, we will be releasing only one expansion per year. We will still continue to provide free content updates throughout the year as well as normal live team maintenance, but plan to use the extra time to create some of the most polished EVERQUEST content the game has seen in a while. ❖

NICHOLAS BELIAEFF is executive director of development at Sony Online Entertainment. Email him at nbeliaeff@gdmag.com.

VICIOUS ENGINE

making games. easier. faster. better.

Engine support for

Nintendo Wii™

PSP® (Playstation® Portable)

PC · XBOX · and more...



www.viciousengine.com

Contact us about our fully
functional 90 day evaluation.

The powerful and flexible technology
behind the highly anticipated
PSP® title: "Dead Head Fred™."

Dead Head Fred™

In stores 8/28/07

*PSP™ is a registered trademark of
Sony Computer Entertainment Inc.

D3PUBLISHER



**VICIOUS
CYCLE®**

BINK



BINK HAS SHIPPED IN 2,900 GAMES FOR A REASON.

IT'S THE STANDARD FOR VIDEO IN GAMES - UP TO 4 TIMES FASTER AND USING AS MUCH AS 18 MB LESS MEMORY THAN OTHER CODECS.

IT HAS A BUILT-IN AUDIO CODEC WITH MULTITRACK CAPABILITY, SUPPORTS EVERY PLATFORM, AND CAN HANDLE AN ARBITRARY NUMBER OF VIDEO CHANNELS, LETTING YOU ADD Z-DEPTH, NORMALS, OR UV PER-PIXEL DATA, AND MORE!

FLAWLESS HD VIDEO WITH 5.1 AUDIO ON WINDOWS, XBOX AND PS3!



Now playing on the Wii, PSP, PS3, and Xbox 360
425-893-4300 www.radgametools.com

