



United Business Media

RESEARCH REPORT: GOVERNMENT INCENTIVES

VOL16NO11 DECEMBER2009

gamedeveloper

THE LEADING GAME INDUSTRY MAGAZINE

flawed beauty

USING IMPERFECTIONS
TO CREATE BELIEVABLE
CHARACTERS

brütal legend

POSTMORTEM



calling all content creators

create opportunities

The Arab world has an estimated 300 million people looking for quality content. twofour54, based in Abu Dhabi, is the region's premier centre for Arabic content creation in the media and entertainment sector across media platforms including television, radio, film, publishing, online, mobile, music, gaming and animation and we can help you reach them.

Create skills

twofour54 tadreeb is developing and enhancing the skills of media professionals enabling them to deliver immediate success to their business.

Create ideas

twofour54 ibtikar provides media and entertainment industry funding and support for start-up and early stage businesses and individuals with creative ideas for Arabic content and a passion for success.

Create content

twofour54 intaj leads the Arab world with fully equipped HD production studios, the latest post-production technologies, media management, digital archiving, playout and uplink services.

Working in partnership with some of the world's major content creation companies, twofour54 provides your business with the right opportunities to grow in the Arab world.

find us. join us. create with us.
destination Abu Dhabi
+971 2 401 2454. twofour54.com

create.

twofour54
Abu Dhabi

content creation community

POST MORTEM

24 DOUBLE FINE'S BRÜTAL LEGEND

BRÜTAL LEGEND is Double Fine's sophomore effort, and like its first title PSYCHONAUTS, was fraught with publisher shifts and new platform adjustments. Here, the team discusses testing bots, lawsuits, metal gods, its use of middleware in conjunction with homegrown tools, and the problem of real time strategy on consoles.

By Caroline Esmurdoc

FEATURES

7 LAND OF OPPORTUNITY

A number of national and regional governments around the world offer tax rebates, grants, and other perks to game developers. In this feature, compiled from a longer *Game Developer Research* article, we outline the major institutions in the Western world that could help you make your next game on the cheap.

By Chris Remo

16 THE DUST OF EVERYDAY LIFE

Modeling characters in 3D is an art in the true sense, and it's quite a challenge to make these characters appear realistic. Takayoshi Sato, who created all the original SILENT HILL CG by himself, finds that adding flaws helps to create something believable—but those flaws can't be random. They must be carefully tied to the character's personality and backstory. Here, Sato shares his thoughts about the creation of compelling characters in games, something more than your average vacant space marine.

By Takayoshi Sato

DEPARTMENTS

2 **GAME PLAN** *By Brandon Sheffield*
Digital Snake Oil

[EDITORIAL]

4 **HEADS UP DISPLAY**
Front Line Award Finalists, TORCHLIGHT and OGRE 3D, and OneBigGame's charity software.

[NEWS]

39 **TOOL BOX** *By Shekhar Dhupelia*
Projecturf's Projecturf.com

[REVIEW]

42 **THE INNER PRODUCT** *By Daniel Nelson*
Fire Control

[PROGRAMMING]

46 **PIXEL PUSHER** *By Steve Theodore*
Unknown Unknowns

[ART]

49 **DESIGN OF THE TIMES** *By Soren Johnson*
Challenging Design

[DESIGN]

51 **AURAL FIXATION** *By Jesse Harlin*
Rallying Rights

[SOUND]

56 **ARRESTED DEVELOPMENT** *By Matthew Wasteland*
Project Status Update!

[HUMOR]



Think Services, 600 Harrison St., 6th Fl.,
San Francisco, CA 94107
t: 415.947.6000 f: 415.947.6090

DIGITAL SNAKE OIL

OF WHAT WORTH ARE VIRTUAL ITEMS?

VIRTUAL ITEMS ARE THE SUBJECT OF MUCH

contention. Are free-to-play games devaluing retail products? Are they changing the industry? Early this month, I was having a discussion about this with Metanet's Raigan Burns, in which he argued that virtual items represent the equivalent of digital snake oil—you're paying for a few altered lines of code.

It's a question of degrees, because all games are lines of code after all, whether they be many or few. And in fact Metanet's latest game, N+, is primarily sold via digital distribution, in many ways a larger, more involved virtual item. But I understand his point very well. The idea of paying money for something that a designer maybe spent an hour tweaking, or which an artist adjusted the colors on just doesn't sit well with me.

This is rooted in our consumer-oriented society. Ultimately all value is perceived. Why is a diamond more valuable than cubic zirconium? Mostly because we say so. As a society we've decided that between these two similar subjects (though the latter is synthetic), one is worth more, and the other less. Meanwhile both are worth more than food, which we actually need to survive.

Food, air, and water have intrinsic value, because we can't live without them. Aside from those stand-out examples, our entire value system is fabricated—so depending on one's desire to have these things, they're worth as much as or more than anything else. It's quite relative, and in a society in which most of us actually do pay for the water we drink, this perception of value is very important to a lot of people, including, I'm dismayed to say, myself.

DR. SHEFFIELD'S CURE-ALL

» For me, if there's an object I can own versus an digital version, I'll go that route every time. I still buy CDs, DVDs, and records, and prefer physical copies of games I really enjoy over digital ones. Over time I'm letting go of this—after all, my enjoyment of these media is not based on their physicality, but rather the data contained on them.

For a lot of people, that need for the physical simply isn't there, and that's why the perceiver is the most important part of perceived value. For someone playing MAPLESTORY who really wants that purple sword because it matches their outfit, that sword is possibly one of the most important things that person could buy.

Raigan's point was this: "Goods like a paperback novel, a pen, or a shovel might have a resale value that's close to zero, but they still have some sort of 'functional' value in that they can be used for some purpose, i.e. I can read or write or dig a hole.

"In comparison, most virtual goods are purely useless. Of course, I'm referring to ANIMAL CROSSING 'cool yellow shirt'-type goods; something like a really good sword in WOW would actually be useful, because it will

allow the owner to farm gold more effectively and then sell the gold on the black market or whatever. But even that is a contrivance, the developer could easily modify a variable to let the player do a lot more damage, they don't "need" the sword—it's an artificial constraint imposed by the developer.

"This is typically benign in 'normal' games because it's done in the service of gameplay, but once you enter virtual goods land though, the rules are designed to extract more money out of people rather than to provide people with an enjoyable experience. This seems very different and possibly awful."

I do agree with Raigan, mostly, and my discussion of perceived value was largely to be contrary—but it's also exactly the reason this model is working. There are people for whom the physical element of the purchase isn't important—they're paying for added fun, and if that fun is in the form of a yellow shirt, so be it.

That's perhaps the most important part—for those who play these games, these items aren't perceived as designed to extract money, they're part of a fun experience. For instance, I'm not a religious person—but what seems to me to be a method of controlling a populace appears to others as a way to approach the divine and achieve personal fulfillment. It's all a question of perception.

GIMME THAT OLDE-TYME RELIGION

» While the concept of paying for something so virtual initially seemed alien to me and my experience, I thought back to good old La Val's Pizza in Berkeley, where I grew up. How many quarters did I scam out of my parents so that I could get a few more lives in FINAL FIGHT, or another go at RAMPART? In essence I was renting time with the game—the virtual items I was paying for were lives. In practice, these free-to-play games that run on microtransactions (subscription or pay-per-play games even more so), which many core or oldschool players decry, extrapolate from a revenue stream that comes from the very source of electronic games.

Like Raigan, I am curmudgeonly reluctant to admit the value of the piecemeal experience over the whole, finished one. But the fact is that more and more people see that free-to-play experience not as piecemeal, not as incomplete, but rather a living experience that can grow and change. Or a new kind of experience with a low required investment.

And some of us fogeys may do well to recall that this model is not so different from that on which we were raised, or for the foye-er amongst us, the games we created. The trick is how to make these virtual items actually worth what the users pay for them. But that's a yarn for another day.

—Brandon Sheffield

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

EDITOR-IN-CHIEF

Brandon Sheffield | bsheffield@gdmag.com

PRODUCTION EDITOR

Jeffrey Fleming | jfleming@gdmag.com

ART DIRECTOR

Joseph Mitch | jtmitch@gdmag.com

CONTRIBUTING EDITORS

Jesse Harlin

Steve Theodore

Daniel Nelson

Soren Johnson

Damion Schubert

ADVISORY BOARD

Hal Barwood Designer-at-Large

Mick West Independent

Brad Bulkley Neversoft

Clinton Keith Independent

Bijan Forutanpour Sony Online Entertainment

Mark DeLoura Independent

Carey Chico Pandemic Studios

ADVERTISING SALES

GLOBAL SALES DIRECTOR

Aaron Murawski e: amurawski@think-services.com
t: 415.947.6227

MEDIA ACCOUNT MANAGER

John Malik Watson e: jmwatson@think-services.com
t: 415.947.6224

GLOBAL ACCOUNT MANAGER, EDUCATION AND RECRUITMENT

Gina Gross e: ggross@think-services.com
t: 415.947.6241

COORDINATOR, EDUCATION AND RECRUITMENT

Rafael Vallin e: rvallin@think-services.com
t: 415.947.6223

ADVERTISING PRODUCTION

PRODUCTION MANAGER

Robert Steigleider e: rsteigleider@ubm-us.com

REPRINTS

WRIGHT'S REPRINTS

Ryan Pratt e: rpratt@wrightsreprints.com
t: 877.652.5295

THINK SERVICES

CEO THINK SERVICES Philip Chapnick

GROUP DIRECTOR Kathy Schoback

CREATIVE DIRECTOR Cliff Scorso

CHIEF INFORMATION OFFICER Anthony Adams

AUDIENCE DEVELOPMENT

TYSON ASSOCIATES Elaine Tyson

e: tysonassoc@aol.com

LIST RENTAL Merit Direct LLC t: 914.368.1000

MARKETING

MARKETING SPECIALIST Mellisa Andrade

e: mandrade@think-services.com

UBM TECHNOLOGY MANAGEMENT

CHIEF EXECUTIVE OFFICER David Levin

CHIEF OPERATING OFFICER Scott Mozarsky

CHIEF FINANCIAL OFFICER David Wein

CORPORATE SENIOR VP SALES Anne Marie Miller

SENIOR VP, STRATEGIC DEV. AND BUSINESS ADMIN. Pat Nohilly

SENIOR VP, MANUFACTURING Marie Myers



IF YOU COULD CREATE A

VIRTUAL WORLD

FOR YOUR BUSINESS

GEORGIA

IS WHAT IT WOULD BE LIKE.

**Produce your next MMOG or virtual world
in Georgia and receive a 30% tax credit!**

Contact: Asante Bradford, abradford@georgia.org 404.962.4056



Visit georgia.org/gamedevelopment or call 877-746-6842



ogre in the torchlight

RUNIC GAME'S RECENTLY- released TORCHLIGHT has been getting favorable notice for its refined gameplay and class-y visuals. Built by an experienced team that includes key talent from the FATE, MYTHOS, DIABLO, and DIABLO II projects, TORCHLIGHT is also noteworthy for its speedy, one-year development time and its use of the free, open source OGRE 3D graphics solution. "Having a mature rendering engine was key to getting TORCHLIGHT completed," Runic president and TORCHLIGHT project lead Travis Baldree told us.

Available under the GNU Lesser Public License, OGRE has been in development since 2001 under the guidance of Steve Streating. The engine provides an object-oriented, cross-platform graphics rendering solution for Direct3D and OpenGL APIs. "We were

attracted to the fact that it was extremely compatible, and it had great support for fixed-function since we eschewed shaders for the most part," Baldree said.

OGRE supports a number of critical features for game visuals including a variety of shadow rendering techniques, particles, skeletal animation, post-processing effects, and an overlay system for building HUDs. "It isn't a game development suite. It's a rendering system, and a damn good one at that," TORCHLIGHT's lead programmer Marsh Lefler told us. "We wanted a renderer that was easy to implement and allowed us to hit the ground running. With OGRE it was a snap to implement and the serious lack of bugs allowed us to do just that."

OGRE was designed with a component-based philosophy

that gives developers the option to integrate its rendering code with any number of libraries. "Just the sheer amount of flexibility we had with OGRE allowed us to create our game and toolsets in parallel quickly and without constant distractions from rendering related bugs," Lefler said.

"We were really pleased with how OGRE handles static geometry and batching. The software skinning is also extremely fast, and the material system is pretty robust," Baldree said. OGRE's design supports a flexible scene graph architecture in which the low level renderer is independent from the scene graph, allowing developers the freedom to use predefined scene organization classes or roll their own.

Free software can sometimes carry the stigma



Runic Games' TORCHLIGHT.

being incomplete or lacking in support, making it unusual for commercial projects to rely on open source solutions. However, Runic found OGRE to be mature and robust. "Our initial concerns had to do with tools and troubleshooting—but we were able to assemble a toolset, and build what wasn't available," Baldree said. "Any kind of issues we did seem to

have were quickly resolved by reading the well documented code or simply visiting the forums. To be honest though, after about two weeks using OGRE those concerns faded away," Lefler said.

OGRE 3D is available free of charge [providing you agree to the open source terms] at www.ogre3d.org.

—Jeffrey Fleming

onebiggame for charity

ONEBIGGAME, A CHARITY-ORIENTED game incubator, has announced its first title (Zoe Mode's CHIME) and named Save the Children and the Starlight Children's Foundation as the beneficiaries of



Martin de Ronde

its charitable contributions. Martin de Ronde, the co-founder of Guerrilla Games, formed OneBigGame over two years ago with the idea of creating a designer "supergroup" that would develop a single epic charity game project. "All the famous individuals—the usual suspects—would get together to produce a mini-game extravaganza. They do a bunch of mini-games, and we have one developer create the combined

game, and that would be it," de Ronde explained. "That was, practically, a little bit ... ambitious. Stick a couple of musicians in a room, and you get a huge single like *Live Aid*. Do it with Will Wright, Peter Dinklage, and Shigeru Miyamoto, and I'm not sure what you get, but you definitely won't have a game."

Instead, the company decided to approach individual developers and designers with the offer of creating more personal, smaller-scale projects not constrained by the typical marketing realities of triple-A game development.

Originally, that meant Flash games, but developers started coming back and asking de Ronde if they could prototype using environments like XNA—and then they started asking if they could simply launch through

console and PC digital distribution platforms rather than the web.

"It was more work for us, but I think in the end, it's going to pay dividends," de Ronde said. "Suddenly, the profile gets raised. Rather than free-to-play Flash games, it's now free-to-play Flash games which will redirect people to the website where you can purchase the full PC version. And if you own an Xbox 360, a PlayStation 3, or a Wii, you can go to these platforms and purchase whatever version is out."

Zoe Mode, a Kuju-owned development studio, was the first OneBigGame collaborator to push for the more ambitious distribution schedule. Its upcoming CHIME, a clever rhythm puzzle game with shades of LUMINES, was developed on a pro bono

basis for Xbox Live Arcade and PC, with a free Flash trial version to accompany the full game.

Going forward, each developer will be able to choose which platforms to target. Additionally, after four to six months of being published under the OneBigName label with profits going to the publisher's charity partners—12 months in the case of Flash games—royalty rights revert to the games' creators, who can then sell their titles however they like.

"I'm really keen to see CHIME do well and become successful, so other developers think, 'I'm going to help too. I'm going to do my bit for charity, and if it's really successful, I might end up with a great new IP.'"

Future game projects are expected from PARAPPA THE RAPPER creator Masaya Matsuura, BROKEN SWORD series director Charles Cecil, and Shiny Entertainment founder Dave Perry.

—Chris Remo

FRONTLINE AWARDS 09



Our annual Game Developer Front Line awards are fast approaching, so we've decided to give you a recap of the finalists in each category; Art, Audio, Books, Engines, Middleware, and Programming/Production. In determining the winners of the 2009 awards we go through a multistep process. Open nominations were held in October, and from that list we consulted with our advisory board to narrow down the results to five entries in each category. We then handed the nominees to over our readers, via an invitational online survey in November, so that our audience would make the ultimate decision. As this issue goes to press, we're compiling the results, and in our January 2010 issue we'll reveal the winners. We'll also be inducting one special game development tool into the Hall of Fame that has been of enduring importance to the development community. Because the editors of *Game Developer* decide the Hall of Fame winner, it is not eligible in the regular categories. Stay tuned next month for the results.

ART TOOL

Autodesk 3ds Max 2010
Autodesk

Autodesk Mudbox 2009
Autodesk

Corel Painter 11
Corel

Modo 401
Luxology

ZBrush 3.12b
Pixologic

AUDIO TOOL

FMOD Designer 4.27
Firelight Technologies

Miles Sound System 7.2e
Rad Game Tools

ProTools v8.0.1
Digidesign

Vivox Precision Studio SDK v. 3.0.6
Vivox

Wwise 2009.2
Audiokinetic

MIDDLEWARE

Havok Physics 6.6.0
Havok

Havok Behavior 6.6.0
Havok

morpheme 2.0
NaturalMotion

Scaleform GFx 3.0
Scaleform

SpeedTree 5.0
Interactive Data Visualization

ENGINE

CryEngine 3
Crytek

Gamebryo Lightspeed
Emergent Game Technologies

Source 1.6.1.6
Valve

Unity 2.5.1
Unity Technologies

Unreal Engine 3 Build 37xx
Epic Games

PROGRAMMING / PRODUCTION TOOL

Adobe Flash CS4 Professional v10
Adobe Systems

Hansoft 5.3
Hansoft

Perforce 2009.1
Perforce Software

SlickEdit 2009 ver 14.0
SlickEdit

XNA Game Studio 3.1
Microsoft

BOOK

Game Coding Complete 3rd Edition
by Mike McShaffry
Charles River Media

Game Engine Architecture
by Jason Gregory
AK Peters

Mastering Unreal Technology: Vol. 1
by Jason Busby, Zak Parrish, and Jeff Wilson
Sams Publishing

Racing the Beam: The Atari Video Computer System
by Nick Montfort and Ian Bogost
The MIT Press

Real Time Cameras: A Guide for Game Designers and Developers
by Mark Haigh-Hutchinson
Morgan Kaufmann

MICHIGAN LOVES GAMERS

CREATE YOUR GAME IN MICHIGAN

The developer/owner of any game created in Michigan is eligible for a rebate of up to 42% of the money spent in Michigan. We're talking cold hard cash.

BUILD A NEW FACILITY IN MICHIGAN

We'll give you a 25% tax credit on building and equipment costs to put it together (or even remodel it).

TALK TO US

We want creative minds in Michigan. We want the gaming industry in Michigan.

Details at www.MichiganFilmOffice.org or call 800.477.3456.

MICHIGAN

F I L M O F F I C E

regional game
development
incentives
around the
world

land of opportunity

AS DEVELOPMENT COSTS FLUCTUATE AND CORE GAMER-ORIENTED PROJECTS

increase in size, developers search for ways to cut costs. One such method is government subsidies, grants, tax breaks, and the like. Regional and national governments have given game and tech companies some sort of compensation for years now, as it helps bring future-looking jobs to the area, and allows the governments to claim a forward-looking strategy. While most grants and tax breaks are for companies of middle to larger size, there are documented cases of indies getting some cash flowing their way as well. Is your company taking full advantage of your government's offers?

In the following pages, we lay out a number of the larger government programs to stimulate regional game development, across North America, the U.K., Australasia, and Continental Europe. We noticed that a number of these regions, especially within the United States, don't have any game developers utilizing their programs. Recently laid off? Thinking of moving back home to Rhode Island? You might be able to take advantage of programs nobody else is using.

While this alphabetical list is not all-inclusive, we feel it represents the largest organizations around the Western world, and should be a good starting point for budding entrepreneurs. May this article help you on your way to more affordable game development!

[More information on this subject can be found in the full report from Game Developer Research at <http://gamedeveloperresearch.com>.]

Land Opportunities

» UNITED STATES

CONNECTICUT: COMMISSION ON CULTURE AND TOURISM

www.cultureandtourism.org/cct/cwp/view.asp?a=2126&q=302556&CCTNAV_GID=1637

The film division of the Connecticut Commission on Culture and Tourism offers a "tax credit equal to 30 percent of qualified digital media production, preproduction, and postproduction expenses incurred in the state that exceed \$50,000" for digital media projects developed for distribution/exhibition to the general public. The production company must be registered with the Secretary of the State of Connecticut.

The Hartford Business Journal report cited below suggests that developers may simply not know about the tax credit's existence, which could explain its relatively low rate of uptake. "It is in the statute and it has always included video games," said Karen Senich, then-acting executive director for the Connecticut Commission on Culture and Tourism. "We've had a lot of interest from the film industry, but not a single application for video games." (www.hbjournal.com/news3553.html)

The program's lack of broad success may end up destroying it—according to WFSB Hartford (www.wfsb.com/money/18694115/detail.html), Venan, the film studio that did take advantage of the credit, is concerned as of February 2009 that state legislators may slash the program in an attempt to tighten the state budget. Still, as of the time of this report's publication, the credit remains.

FLORIDA: THE GOVERNOR'S OFFICE OF FILM AND ENTERTAINMENT

www.filminflorida.com

In addition to various production-related discounts, the Governor's Office of Film and Entertainment has a 10 percent cash rebate program (rather than a tax credit) for eligible interactive entertainment productions,

provided they meet "the required minimum qualified expenditures of \$300,000 on a single qualified project." The maximum reimbursement on a single project is \$1,000,000. A company may not apply for more than three projects in any one fiscal year.

At least 50 percent of employees hired for the digital media project must be Florida residents. Costs incurred for marketing, publicity, distribution, and the conceiving/storyboarding of the videogame are not considered qualified expenditures.

In total, the Governor's Office says that Florida hosts 383 entertainment production companies employing some 3,400 workers.

According to *Game Developer Research's* Game Developer Census 2008, Florida is one of only seven states to employ more than 1,000 game development professionals.

GEORGIA: DEPARTMENT OF ECONOMIC DEVELOPMENT

www.georgia.org/GeorgiaIndustries/Entertainment/VideoGames/Pages/default.aspx

The Georgia Entertainment Industry Investment Act provides digital entertainment producer—including video game companies based outside of the state—tax credits on products developed there.

Qualifying game developers or publishers are eligible to receive a 20 percent base tax credit on all qualified expenditures within Georgia, such as labor, materials, and services, as well as an additional 10 percent tax credit if the client is willing to place a Georgia promotional logo within the video game product. An additional incentive provides for a Sales and Use Tax Exemption of up to 8 percent on all qualified expenses.

In addition to its statewide benefits, the city of Savannah, by way of Georgia's Creative Coast Alliance and the Savannah Economic Development Authority, is looking to aggressively build its local game development presence by offering free rent for a year within its Game Development and Digital Media Center.

Since the release of last year's Government Incentive Report, the state of Georgia has significantly boosted its game development benefits. In a statement made at the time, it was reported that the new incentives were aimed at growing the financial impact of Georgia's overall entertainment industry, which had been declining since its \$1.17 billion peak in 2007.

According to Georgia Department of Economic Development Commissioner Ken Stewart, "The new incentives will put Georgia among the top five states in the U.S. in terms of financial competitiveness for entertainment projects. We expect to see an increase in the number of industry jobs and overall economic impact for the state in the coming years."

HAWAII FILM OFFICE

www.hawaiifilmoffice.com

This is a refundable tax credit based on a production company's Hawaii expenditures while producing a qualified film, television, commercial, or digital media project.

The credit equals 15 percent of qualified production costs incurred on Oahu, and 20 percent on the neighbor islands (Big Island, Kauai, Lanai, Maui, Molokai).

Hawaii has few game developers, and as a result there seem to be few publicly-known cases of video game companies taking advantage of its incentives for digital media productions.

According to a 2004 report by The Honolulu Advertiser (http://findarticles.com/p/articles/mi_qn4176/is_20040425/ai_n14572069), the state's entertainment industry is substantially dominated by film productions, and the official website reflects that focus. At that time, The Advertiser reported the state's game development jobs to number fewer than 200. Since that time, Konami closed its Honolulu office, which employed around 40 workers.

Still, Honolulu serves as base of operations for TETRIS co-controller Henk Rogers, who heads up various TETRIS-related firms as well as his development company Avatar Reality, which is currently developing the ambitious MMO and online platform BLUE MARS.

LOUISIANA ENTERTAINMENT

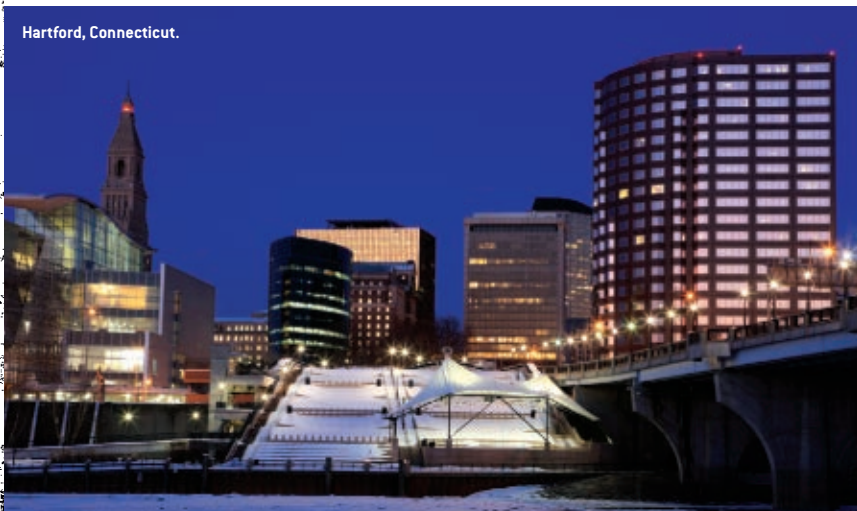
www.louisianaentertainment.gov

Louisiana gives game developers a tax credit of 25 percent, with an additional 10 percent credit given on the payroll taxes of in-state resident hires. There is a carry forward on those tax credits earned for ten years. There is no cap on the tax credits earned.

Since the publication of our 2008 Government Game Incentive Report, Louisiana has appreciably boosted its total incentive to game developers with the Digital Media Act, raising the tax credit from 20 to 25 percent, including a 10 percent state payroll tax credit, and removing the graduated nature of the previous credit.

Asked by *Game Developer Research* to highlight key positive elements of the state's offerings for game developers, Louisiana Entertainment asserted that Louisiana is "specifically targeting and supporting

Hartford, Connecticut.



game development”—a claim that seems to be backed up by its recent Digital Media Act boost. [The public-private BRADIC partnership, which covers the Baton Rouge area, is one of the areas within Louisiana that particularly advocates for game developers to relocate or set up in the immediate surrounding area.]

The office also cited its general low cost of living and low mortgage and interest rates, as well as the availability of “technology business incubators built for the purpose of incubating small technology businesses such as small game development studios.”

MAINE FILM OFFICE

<http://filminmaine.com>

The Maine Attraction Film Incentive Package, administered by the Maine Office, includes a wage reimbursement plan that reimburses “10 percent of the amount paid as wages for non-Maine residents and 12 percent of the amount paid as wages for Maine residents.”

A Certified Media Income Tax Credit also offers a “non-refundable credit equal to the Maine income tax otherwise due on taxable income related to the certified media production.” The production company must spend \$250,000 on in-state production-related costs during a 12-month consecutive period, but is not required to budget that amount to one single project.

There are extremely few game developers operating in Maine. Battlefront.com of Dover-Foxcroft appears to one of the only full-fledged developers in the state. Digitalmill of Portland operates a number of games industry-related conferences, particularly surrounding serious games, including Games for Health.

MICHIGAN FILM OFFICE

www.michiganfilmoffice.org

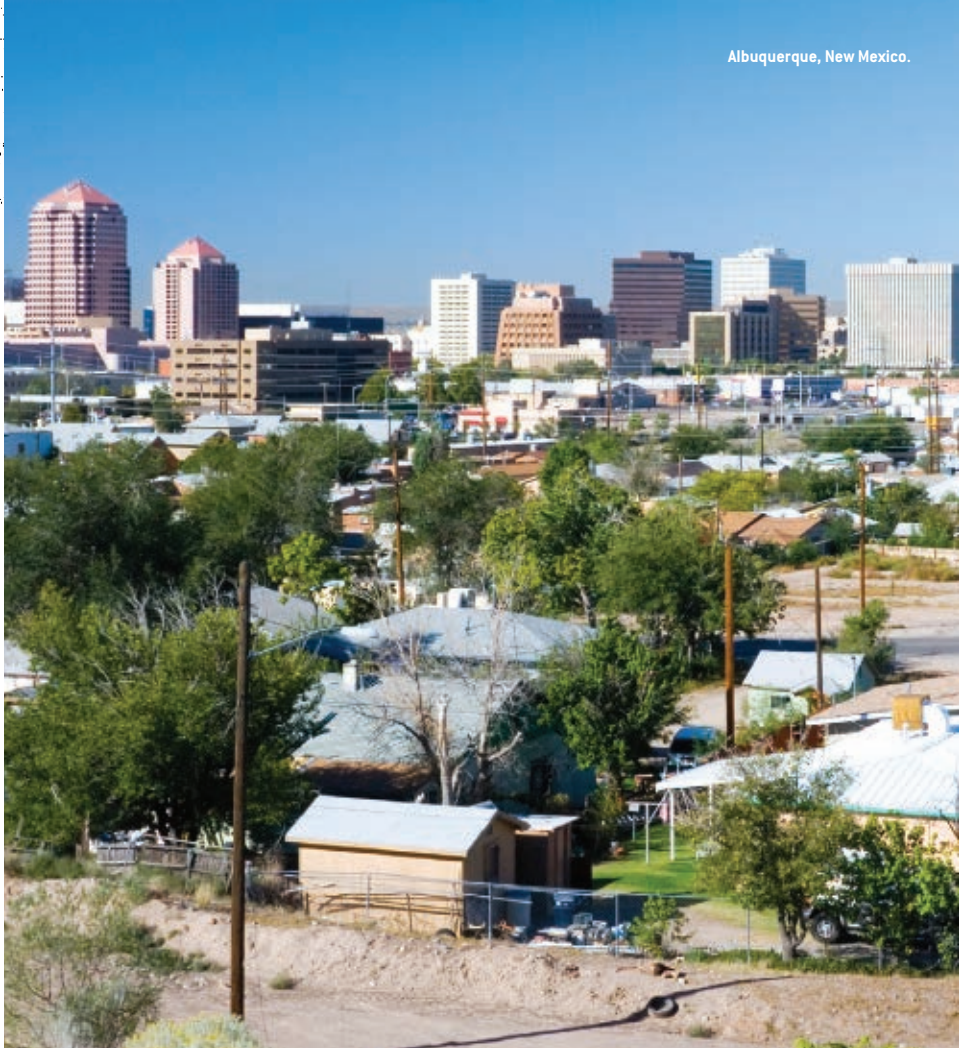
Michigan’s film production credit, which also applies to video game productions, is a tax credit of 40 percent (up to 42 percent depending on the type of expenditures) on production spending within the state. The payroll credit is only 30 percent for non-resident workers.

Projects with an in-state budget of at least \$50,000 are eligible. There is a \$2 million salary cap per employee per production, with no other caps in effect.

Michigan’s benefits, primarily aimed at the film industry but also applicable to video game products, were passed last April, and are among the most generous in the country.

Said Michigan governor Jennifer Granholm, who specifically cited Stardock in her 2009 State of the State address, “The fact that these jobs exist in Michigan today is no accident. These jobs are here because we put a strategy in place to bring them here—often by beating out other states and other countries to get them.” (<http://blog.mlive.com/manzero/2009/02/michiganbasedstardocksystems.html>)

According to statements later made to *Game Developer Research* following that announcement, the benefits were a combination of the available production credits as detailed in this report, and



specific additional infrastructure promises that would allow Stardock the proper network support for its current massively multiplayer project.

NEW MEXICO FILM OFFICE

www.nmfilm.com

New Mexico offers a 25 percent Tax Rebate on all direct production expenditures that are subject to taxation by the State of New Mexico. It applies to feature films, independent films, television, regional and national commercials, documentaries, video games, and post-production.

New Mexico appears to have extremely minimal game developer presence. A small independent studio, Tenth Artist, was formed in Albuquerque out of the team that developed CHROMA GEIST for PC and Xbox 360, but little information is available.

The state does have its own International Game Developers Association chapter, IGDA Albuquerque, and held its own local NM Game Jam development event, but records of currently operating full-scale development seems essentially nonexistent.

RHODE ISLAND FILM & TELEVISION OFFICE

www.film.ri.gov

The Rhode Island Film and Television Office provides state-certified video game productions a 25 percent tax credit for preproduction, production, and

postproduction costs incurred in Rhode Island. There is no cap, but the production must have a minimum \$300,000 budget for costs directly attributable to activity within the state. The production must be created primarily in the state of Rhode Island.

Rhode Island appears to have minimal commercial game development, despite a number of game development-related degree programs offered at several art- and design-focused schools located within the state.

TEXAS FILM COMMISSION

www.governor.state.tx.us/film/industries/games

Video game productions can receive an incentive payment at the value of 5 percent of eligible spending in the state of Texas.

To qualify, the in-state spending must be at least \$100,000, 60 percent of the production time must have been spent in Texas, and 70 percent of paid employees and contractors must be state residents.

In April 2009, Texas passed new legislation updating its incentive offers for video game developers. While the basic terms of the incentive—a 5 percent payment on Texas spending—have not changed, there was a significant change in the removal of the total payment limit, which as of the publication of the previous Government Game Incentive Report was capped at \$250,000.

Texas, and Austin in particular, is one of the United States’ major development hubs, with a strong history

Canada Opportunities

in the shooter and massively multiplayer segments. *Game Developer Research's* Game Developer Census in 2008 found Texas to be the third largest development community in the nation with a total of about 3,000 workers in the field.

WISCONSIN: FILM WISCONSIN

www.filmwisconsin.net

Film Wisconsin offers a 25 percent state income tax credit to electronic game production businesses with accredited projects that meet the minimum \$50,000 budget requirement for in-state production-related costs and wages of Wisconsin residents. Video game projects are eligible for a development period of 36 months. A project is eligible if 35 percent of its total budget is spent in Wisconsin.

Wisconsin is not one of the United States' major game hubs, but its Madison area does host a notable cluster of studios, some of which have grown out of the presence of two-decade-old Raven Software. As such, there has been a good deal of reporting over the last several years speculating about the potential for Madison to become a top-ranking game development center.

In 2008, the state boosted its tax breaks for game developers, instituting an impressive 25 percent credit. However, in 2009, Governor Jim Doyle vetoed a number of specific aspects of the incentive, keeping its basic credit of 25 percent but partially limiting the overall scope.

Under the new rules, nonresident salaries get no tax credit, whereas they previously got a lower credit than residents. A bonus 3 percent credit for working in "economically distressed areas" was also removed. The overall funding for the incentive bill was also significantly decreased from \$1.5 million to \$500,000.

Still, there were silver linings as well. The production eligibility threshold was lowered from \$100,000 to \$50,000, and the length of eligible duration for video game productions for the application of the incentive was raised from 12 months to 36 months.

>> CANADA

TÉLÉFILM CANADA

www.telefilm.gc.ca

The Canada New Media Fund, administered by Telefilm Canada, has a total of \$14.3 million dollars (CAN) allocated per year for a period of two years by the Department of Canadian Heritage. The applying company must be Canadian-owned and controlled, having a head office based in Canada and owning 100 percent of the product's copyright. At least 75 percent of the production's eligible costs must be incurred in Canada.

Telefilm Canada's grants come in the form of "conditionally repayable advances"—certain terms can reduce the amount a developer must repay. For example, releasing a game with both English and French language options will reduce the repayable amount by 10 percent.



Manitoba, Canada.

Many Telefilm Canada-funded projects are better described as general interactive media rather than "video games" in the sense of the video game industry, but there are plenty of game projects that have taken advantage of the grants as well.

There is a definite tilt toward lower-budget titles, and those with an "indie" flavor, but larger scale projects like Hothead Games' DEATHSPANK are sometimes able to secure grants of proportionally higher worth.

MANITOBA: MANITOBA SCIENCE, TECHNOLOGY, ENERGY, AND MINES

www.gov.mb.ca/stem/knowledge/digital_media.html

The product assistance portion of the Manitoba Interactive Digital Media Fund "provides recoupable financial support for the market research and prototyping, product development and marketing of Interactive Digital Media works that are intended for the general public and are created, owned and controlled by Manitoba companies." Manitoba Science, Technology, Energy and Mines will typically provide up to 50 percent of approved costs for market research and prototyping, to a maximum of \$20,000.

The fund also offers non-repayable advances "for the professional development and/or market attendance of Manitoba individuals and private companies actively involved in Interactive Digital Media prototyping, product development and marketing." Up to 50 percent of approved costs, to a maximum of \$50,000, for this phase will be eligible.

The industry assistance component of the fund also offers non-repayable advances that "financially support activities and initiatives that contribute to the development of the Interactive Digital Media industry in Manitoba." Up to 50 percent of approved eligible costs, to a maximum of \$25,000, will be covered.

Separately, Manitoba Science, Technology, Energy, and Mines offers its Interactive Digital Media Tax credit equal to 40 percent of wages paid to Manitoba residents on eligible projects, up to a maximum credit amount of \$500,000.

Compared to some other Canadian provinces, Manitoba is not a particularly strong game development hub, although there are a number of studios located in Winnipeg. The province has also become increasingly aggressive with its game development incentives.

The Fortune Cat Game Studio incubator (www.fortunecatgames.ca) also offers a relatively unique opportunity. Though it received government assistance in 2006 upon its creation, it is not a government initiative. It provides teams physical space and support to conceive games, and gives them funding and assistance in starting their own studios to produce prototypes and eventually full games. It bills itself as the first such incubator in North America. It appears to be operational, having kickstarted several teams.

NOVA SCOTIA: DEPARTMENT OF FINANCE

www.gov.ns.ca/finance/en/home/taxation/businessstax/corporateincometax/digitalmediataxcredit.aspx

The Nova Scotia Department of Finance offers a "refundable tax credit for costs directly related to the development of interactive digital media products." The Nova Scotia Digital Media Tax Credit provides corporations with a credit equal to "50 percent of eligible Nova Scotia labor expenditures" or "25 percent of total expenditures made in Nova Scotia" (whichever is lesser). An additional 10 percent bonus on "labor expenditures" (5 percent on total expenditures) is available for production development outside of the Halifax Regional Municipality. There is also a credit for marketing and distribution expenditures (max \$100,000 per production).

The production company must be a taxable Canadian corporation with a "permanent establishment in Nova Scotia."

Nova Scotia currently has fairly minimal game development presence. HB Studios is its only full-fledged video game developer. The Lunenburg-headquartered firm last year opened a second studio in Halifax.

Reliable Pathfinding Technology. Havok AI.



havok[™]
www.havok.com

*Turning Creative Aspirations
into Technical Realities[™]*

and opportunities

Since *Game Developer Research* last compiled this information, the Nova Scotia Department of Finance stepped up its available support for game developers considerably. Nearly every benefit offered through its tax credit increased, and some of them doubled, such as additional bonuses for non-Halifax labor and total expenditures. The base credit of 35 percent for local labor expenditures or 17.5 percent for total expenditures also increased to a substantial 50 percent and 25 percent respectively.

ONTARIO MEDIA DEVELOPMENT CORPORATION

www.gov.ns.ca/finance/en/home/taxation/businessstax/corporateincometax/digitalmediataxcredit.aspx

The Ontario Interactive Digital Media Tax Credit is a refundable 40 percent tax credit based on "eligible Ontario labor expenditures and eligible marketing and distribution expenses," or 35 percent if the game is a "specified project" developed under a "fee-for-service arrangement." Marketing and distribution expenses are capped at \$100,000 for each production, but there is no limit on "the amount of eligible Ontario labor expenditures which may qualify and there are no per project or annual corporate limits on the amount of the tax credit which may be claimed."

The production company must be a Canadian corporation with a "permanent establishment in Ontario," filing an Ontario tax return.

Separately, the Media Development Corporation

maintains the Interactive Digital Media Fund, which provides non-refundable contributions of up to \$150,000 (capped at 50 percent of the project's budget) to "provide Ontario interactive digital media content companies with access to the final piece of funding required to move their content projects into production."

Ontario is home to Toronto, one of the major Canadian development hubs. Still, the province does not have quite the same standing it desires in the video game industry, and it has been aggressively increasing its incentives for game developers year by year. Since the last Government Game Incentive Report in 2008, Ontario has boosted its total tax credit to 40 percent from 30 percent, and has removed the distinction between small and large businesses that previously gave bigger bonuses to small businesses. That was already an increase from the previous incentive of 20 percent, which was raised in 2006.

Ubisoft's recent expansion into Toronto may also represent a significant step for the province. Ubisoft as well as other major publishers like Electronic Arts already have a big presence in other Canadian cities like Vancouver and Montreal, but Toronto has traditionally been home to smaller or independent studios.

PRINCE EDWARD ISLAND BUSINESS DEVELOPMENT

www.peibusinessdevelopment.com

Technology PEI has in place a Progressive Tax Rebate Program which includes an Innovation and Development Tax Credit for "interactive, information, and

communications technology" firms. It is a refundable tax rebate "equal to 35 percent of eligible salaries and wages" (although that amount is "subject to a 150 percent increase for overhead and other necessary costs").

Short-term productions are encouraged—development may exceed two years, but only two-year expenditures are eligible for the tax credit. The production company must have a "permanent establishment in Prince Edward Island."

In particular, there is concrete evidence of Prince Edward Island's government taking an extremely active role in promoting local game development in the case of Longtail Studios, a small mobile-focused developer in Charlottetown which is being heavily courted by the government of Nova Scotia in an attempt to encourage the company to relocate to Halifax.

According to statements made by the province's Innovation Minister Allan Campbell to CBC News (www.cbc.ca/canada/nova-scotia/story/2009/07/07/pei-gaming-company.html), Prince Edward Island is pursuing additional concrete financial benefits to persuade the company to remain. "I am concerned with the possible loss of these positions on P.E.I.," he said. "I've asked staff in my department to put together a package that is attractive to the company and that invites them to remain here on P.E.I.," he said.

What is particularly notable is that Longtail's Charlottetown location employs only 23 people—demonstrating the provincial government's particularly determined attitude in spurring and retaining a strong local game industry presence.

QUEBEC: MINISTRY OF REVENUE OF QUEBEC

www.revenu.gouv.qc.ca/eng/entreprise/impot/societes/credits/adaptation/multimedias.asp

The Ministry of Revenue of Québec offers a tax credit for labor expenditures to qualified corporations producing multimedia titles. Productions that are commercialized and available in French will receive a 37.5 percent credit, whereas commercial titles not available in French receive a 30 percent credit, and other titles receive a 26.25 percent credit.

The tax credit for corporations specialized in the production of multimedia titles is for "corporations whose activities consist exclusively or almost exclusively in the production of multimedia titles. The credit is granted with respect to the qualified labor expenditure incurred by the corporation for eligible production work."

Along with British Columbia (thanks to Vancouver), Quebec boasts one of Canada's two major video game development hubs in Montreal, with a secondary concentration of developers in Quebec City. Vancouver is generally seen as the country's premiere development location, but many estimations peg Montreal as ready to overtake its competitor to the west.

In August 2009, *The Globe and Mail* reported the Montreal game development industry employs roughly 4,400 workers at more than 40 companies, with Quebec City employing roughly 600 people at five companies.



Toronto, Canada.

Isle of Man.



Quebec has been offering aggressive tax breaks for the industry for well over a decade, steadily solidifying the Montreal game industry, chiefly centered around Ubisoft. The French publisher alone accounts for nearly half of the province's game workers, with 2,200 employed currently and a goal of adding 800 more within four years.

There is some controversy within the industry, however, about Quebec's heavy incentivizing. Speaking to Gamasutra, Silicon Knights founder Denis Dyack (whose studio resides in Ontario) called the province's subsidies "insanely good" and predicted that such significant handouts are not sustainable; furthermore, he argued that the practice results in a "brain drain" taking talent away from elsewhere in Canada.

In a Develop editorial (www.develop-online.net/features/151/Tax-breaks-panacea-or-problem), Mark Greenshields, who ran a Montreal-based studio for seven years, argued that Quebec's tax breaks—while once highly attractive—have created a huge number of Montreal-area developers but not an equivalent amount of talent. He believes that, now, studios in the city must "spend a lot of money developing your games using bigger teams than are necessary, a situation created partly by the tax credit system and partly by the speed of growth in Montreal"—and he alleges rapid salary increases in the area have effectively canceled out the benefit of the tax breaks in the first place.

» UNITED KINGDOM

TECHNOLOGY STRATEGY BOARD

www.innovateuk.org

The Technology Strategy Board is an "executive non-departmental public body" created by the U.K. government and spun off from the Department of Trade and Industry—now the Department for Business Innovation and Skills. With a funding of about £200 million per year, it provides various types of support to companies operating in several targeted technology sectors including, as of last year, computer games.

The U.K. has long been criticized by its local game industry for its lack of significant video game developer-specific tax breaks, particularly compared to Canada's notoriously generous incentives, as well as those of several U.S. states.

UK trade group Tiga in particular is one of the strongest voices, repeatedly calling for more U.K. industry incentives. "Our industry faces challenges that can no longer be ignored: in particular, heavily subsidized competition from abroad and skill shortages at home," said Tiga CEO Richard Wilson according to a GamesIndustry.biz report, adding that the group will "argue for a tax break for games production and for measures to improve the quantity and quality of graduates relevant to the games industry."

A more recent report suggests that the U.K. government may finally be taking such complaints to heart. In the meantime, programs such as the Technology Strategy Board do now offer assistance on a case-by-case basis.

EAST MIDLANDS: EM MEDIA

www.em-media.org.uk

EM Media typically invests up to 50 percent of a production's total cost with expectations of financial and other returns. "Production co-finance for features and large scale projects will be up to a maximum of £250,000." Applicants must be either residents or have a trading address in the East Midlands region. EM Media will not invest in "projects in which EM Media's finance becomes pure subsidy" or individuals/organizations that have already "received over £200,000 worth of public sector support in the past three years."

EM Media has a long history of significantly assisting its local video game development community, much of which has sprung up around the success of Rare, a fixture of the area since the early 1980s. EM Media has invested in both particular game projects and the construction of entire studios.

While East Midlands is not a major worldwide development hub, it has several well-respected studios with a relatively long history, as well as smaller, younger firms. According to the Sector Skills Council for Creative Media, the East Midlands video game industry employs 400 workers.

In its 2007/2008 reporting year, EM Media said it invested £400,000 across five game-related projects, which totaled £3.9 million. Three of those investments were for specific productions, and two were for "business support." The organization reports its expectation that it will recoup its game-related investments.

ISLE OF MAN: DEPARTMENT OF TRADE & INDUSTRY

www.gov.im/ebusiness

Isle of Man's E-Business Division has a Financial Assistance scheme which offers 40 percent equipment grants, 40 percent building grants, 40 percent revenue grants, 50 percent training grants, and more. The country's Business Support Scheme also offers a

grant of 50 percent toward the cost of projects in a wide range of business disciplines up to a maximum level of assistance of £4500 per project.

Although the Isle of Man has made an ongoing concerted effort to attract video game developers, the company's game development industry currently revolves almost completely around online gambling—a segment that has thrived on the island due to strong network infrastructure, zero rate corporate tax, a low betting duty on gross gaming yield, and strong government assistance. According to an Isle of Man Today report published in July 2009, that industry spent £91 million on the Isle of Man in 2008, with a projection of £118 million spent in 2009.

Still, the island has aggressively courted the video game industry, and the government has frequently sent representatives to trade shows such as Game Developers Conference in an attempt to drum up developer interest.

» EUROPE

EUROPEAN COMMISSION: MEDIA PROGRAMME

http://ec.europa.eu/information_society/media

The MEDIA program's "support for the development of on an off-line interactive works" grant is worth a minimum of 10,000 euro and a maximum of 60,000 euro except for the development of "prototypes for games consoles, handheld consoles, and computers." The maximum grant for those productions is 100,000 euro. The applying company must hold the copyright of the production.

Following a legal inquiry, the European Commission also approved France's policy of providing 20 percent tax breaks to French-owned video game developers and publishers.

The European Commission has only been offering video game grants for two years. Interestingly (and encouragingly for the video game industry), according to *Game Developer Research's* calculations, while the

and opportunities

number of grants awarded has remained at a total of fifteen in each year, the total value of grants awarded has risen a substantial 25 percent to 1.23 million euro in only one year.

Furthermore, the proportion of grants awarded at the maximum 100,000 euro level increased from only 20 percent in the 2008 year to 73 percent in the 2009 year—a massive increase.

BELGIUM: FLANDERS AUDIOVISUAL FUND www.vaf.be/taal/en

The Flanders Audiovisual Fund (Vlaams Audiovisueel Fonds) supports the audiovisual community in Flanders by financially supporting productions, including in some cases video games. Of the fund's annual 12.5 million euro governmental grant, at least 78 percent—9.75 million euro—is for supporting projects.

The Flanders Audiovisual Fund does not explicitly support games, but indie studio Tale of Tales has demonstrated that it is possible to obtain funding for games under the Fund's "experimental media" category. The Fund is heavily slanted towards artistically-motivated projects (as opposed to purely commercial ones) in this area, meaning many game developers may have difficulty succeeding.

NORWEGIAN FILM INSTITUTE www.filmfondet.no/iCM.aspx?PageId=118

The Norwegian Film Institute invests in various types of Norwegian-produced or co-developed audiovisual media projects, including video games. As of 2007, its appropriation was NOK 292 million. No individual project may receive more than NOK 30 million in support.

The investment is repaid at a rate of 35 percent once net sales reach the amount of 30 percent greater than the original equity investment.

The Norwegian Film Institute has existed since 2001; in 2004, it began investing in video games in addition to films and television.

Norway's video game industry is relatively small, but is growing rapidly. According to figures from PricewaterhouseCoopers reported by Norway's Ministry of Culture and Church Affairs in 2008, game sales rose 130 percent from 2003 to 2006. Developer and publisher Funcom, which has a particular focus on massively multiplayer games, is the primary driver of these sales; in that same time frame, its share of the Norwegian game market was reported to have increased from about 83 percent to 90 percent.

According to data published by the Ministry of Culture and Church Affairs, the Norwegian Film Fund has awarded between six and nine grants to video game projects per year since 2004, with total support ranging from 5 million NOK to 8 million NOK. Interestingly, the total number of grant applications (and thus awarded grants) has somewhat declined in that time period.

NORDIC GAME PROGRAM <http://nordicgameprogram.org>

The Nordic Game Program supports independent (non-foreign, non-publisher-owned) game developers in Denmark, Faroe Islands, Greenland, Finland, Åland Islands, Iceland, Norway, and Sweden. In 2009, the program was granted DKK 6 million. Investment per project has a minimum of DKK 100,000 and a maximum of DKK 600,000, not to exceed 75 percent of the project's budget.

Though it is one of the newest organizations providing support to game developers, having only been created in 2006, the Nordic Game Program is also one of the most active and aggressive, representing several countries and frequently turning up at industry events worldwide to showcase its region's achievements.

Established as a six-year program by the Nordic Council of Ministers, the Nordic Game Program is in its third year of funding.

In addition to its funding initiatives, the program also operates additional resources for the Nordic game development community, including the extensive directory NordicGame.net and its own Nordic Game Conference, which also attends many other conferences.

Despite the program's success—or, perhaps, partially because of it—program manager Erik Robertson said in April 2009 it is becoming increasingly unable to deal with the volume of submissions it believes are qualified to receive funding.

FINLAND: TEKES www.tekes.fi

Tekes offers either low-interest loans or grants, "depending on the stage of the innovation and the nature of the proposed project." Verso, Tekes' Vertical Software Solutions Technology Programme "supports the development of customer-sector-specific clusters to enhance collaboration between software businesses and sectors deploying the software." Production companies in Finland can apply for Tekes funding at any time.

Despite the country's small size, Finland has a strong, well-regarded video game development community, with a number of companies responsible for worldwide hit games. Tekes provides the industry with strong support, specifically for prototyping and technology development. According to a Develop report published in July 2009, Tekes invested in the Finnish game industry a total of 10 million euro in 2008, deeming the segment as "strategically important." The Finnish game development community is estimated to include about 50 companies.

» AUSTRALASIA

AUSTRALIA: FILM VICTORIA <http://film.vic.gov.au>

Film Victoria provides one-off funding "to develop a prototype of a digital media project." The group provides funds of up to \$500,000 (AUD) for home

console projects, up to \$150,000 for PC and handheld projects, and up to \$50,000 for mobile projects.

In addition to prototype funding, it offers investment of up to \$150,000 to supplement existing funds (such as those from a publisher) for a developer to complete a project. Furthermore, funding of up to \$5,000 is available to companies in the early stages of development for "scoping the technical, creative, market and audience potential of a digital media project."

Finally, a separate organization, Multimedia Victoria (www.mmv.vic.gov.au/InvestInVictorianICT) offers video game developers "free and confidential services in collaboration with Invest Victoria, designed to make establishing or expanding your business in Melbourne and Victoria as straightforward as possible."

Victoria houses one of Australia's two most prolific game development communities (along with Brisbane, Queensland), primarily residing in the city of Melbourne. According to the Australian Government Culture Portal as of 2007, about half of Australia's video game development professionals (across approximately 40 companies) work in Melbourne.

Film Victoria is quite active and consistent in its support of its local video game industry, as can be evidenced by delving into its funding records over the last few years. The government also has a history of proactive behavior—in 2002, Gamasutra reported that Multimedia Victoria was working with the Game Developers Association of Australia to provide PlayStation 2 development kits to Australian developers, since Sony had no office of its own in the country.


NEW ZEALAND: TECHNZ www.frst.govt.nz/funding/business

TechNZ awards up to NZ\$50 million for businesses operating in a variety of targeted technological industries, \$40 million of which is available for "technology funding" including—potentially, but apparently rather rarely—video games (characterized as "information and communication technologies").

Recently revamped, the program now operates through a number of regional advisers rather than as one centralized office. These TechNZ partners provide grants of up to \$100,000.

New Zealand, being a relatively small nation, is not a major video game development center, but its most well-known developer, Sidhe Interactive, has become increasingly high-profile and successful over its decade-plus history.

As far as *Game Developer Research* has been able to determine, Sidhe Interactive is the only major firm to date to have received funding from TechNZ—the fact that it is looking to create a cross-platform development framework, as opposed to seeking funding for a single game project, is likely to have been a big part of why. While many of the other organizations in this report have programs aimed at supporting individual projects, TechNZ does not explicitly target games, and is focused more on the broader technology industry. Sidhe's pitch likely seemed appropriately far reaching to fall under the organization's purview. **PD**



EVEN IF YOU
GOT YOUR
OFFICE CHAIRS
FROM THAT
DUMPSTER
BEHIND THE
BUILDING,

your game can be Unreal.

No matter what size your budget. No matter what type of game.
Unreal can be your game engine. Email Mark Rein at getunreal@epicgames.com



the art of
building
characters

the dust of every day life

TECHNOLOGY HAS GREATLY IMPROVED

since the first 3D consoles appeared. Back then, as the more veteran artists may recall, one building was less than 50 triangles, and a character was less than 500. The standard for a texture was 16 colors at 128 X 128 resolution. It was extremely challenging to render anything organic. We needed to work within severe limitations, and players had to accept pixelated textures all over the place.

Now we've got higher console and PC specs, and the luxury of technologies such as normal, height, and specular maps, ambient occlusion, sub surface scattering, and lots of real-time rendering features that can compete with software rendering. I recall around 2005, close to the release of the Xbox 360, we started hearing about the Uncanny Valley, and the term has only gained traction since then. This means despite the fact that rendering quality has improved tremendously and characters are rendered with high detail, they still lack emotion, and our extremely detailed models are acting like marionettes. The better rendered the model is, the weirder it tends to look.

3D scanning and motion capture cannot be completely trusted either. Motion capture has its own odd look, and 3D scanned heads often don't look anything like the original person. Partially this is because some sort of distortion occurs in the process, but more than that, we don't get the feeling of a person from that frozen still head, like we do not get that feeling of humanity by looking at decapitated head separate from its body. Once I was involved with a very complex experimental scanning process, scanning a 3D shape at 60 frames per second, while recording texture and voice at the same time. It should have captured all human



FIGURE 1 Angela from SILENT HILL 2.

essence in one shot. It should have looked very lively. But it did not.

These soulless models playing out actions before us leave us with an empty feeling. Characters need to be emotionally engaging, touching, and must dominate the scenes they're in with their presence. This cannot be done with any type of recording. Character creation spans several areas outside of the character department (in a typical game production)—in other words, a solid character cannot be established without many disciplines working together as a team, managed by a strong vision. It needs to be elaborately planned, because creating attractive characters is essentially the same as creating good scenes or stories.

In this article I will try to break down the most important elements involved in the creation of an emotionally involving character.

KEY-SHOT CREATION

» A good story consists of several key moments connected together, and characters exist as a vehicle with which to navigate these key moments. So the first thing to do is create and understand your key moments.

There are two prominent shots that we need to pay extra attention to in regard to character. The first is the introduction shot. This is literally an introduction to the character, and provides a first impression for players. Second is the reaction shot, focusing on a character after

a given important event happens. It shows the character's face, telling the audience what kind of incident it was for the character, which helps to define their position within the story. Usually in linear media the scene ends with this reaction shot, and it remains in the audience's mind as a major story impression, whether they are conscious of it or not.

MOOD AND COMPOSITION

» You cannot really do anything without understanding and determining the mood. A simple character description is just not good enough. A figure could look totally different depending on the mood surrounding it.

For instance, a smile that looks like a horrifying grin in Rembrandt



FIGURE 2 Here are photos in which all obvious visual language has been hidden. Makeup is another example of visual language. For instance, almost every one of Marilyn Monroe's facial features has become iconic.

FIGURE 3 The woman on the right has been given a stretched mouth to reflect her deeper character.

lighting might appear innocent with a Renoir lighting scheme. Or if the subject is a monster, with a Brueghel-like background it may look like a legendary story, but would appear to be more of a surreal fantasy in Odilon Redon's style.

Core elements such as story, theme, and philosophical message determine the mood of the target scene. On a more subliminal, but no less important level, lighting and composition are main contributors that define this mood visually. Characters can look totally different under different a lighting scheme, lens, or scene composition. Important shots needs to be designed with those elements in mind, especially in cut scenes or establishing/reaction shots. Ultimately we are aiming for that perfect shot, and character development ideally starts with the scene test bed with the lighting and camera prepared.

In the image of Angela from SILENT HILL 2 [see Figure 1.] there is not much obvious facial expression or any emotional depiction on the character, but you can tell by the subtle things like lighting and the camera setting that there is enough room behind her for somebody to attack, and there is someone [the player in this case] in the scene—and she knows, but does not care much. There is very stable screen construction following her bodylines, yet the banked camera conveys a somewhat unstable moment. She seems to be indulging in her own moment defenselessly. Her eyes tell us there is nothing important in the direction she is looking other than the knife, but her eyes don't

focus on it. This image is giving us a lot of information without using the typically overt emotional tools seen in games.

PERSONALITY DESIGN

» In early game development, pre-production art tends to start with various designs for vehicles, space ships, battle suits, and things of that nature. People tend not to be satisfied or confident with those early designs, and pre-production often gets excessive. In regard to character, combat suits and the like are the industry's favorite theme, and they're put through hundreds of designs as though product sales depended entirely on that. But how much of that pre-production time is spent on what's inside?

In approved concept art, usually we see fantastic costumes, accessories, and cool tattoos. But

the drawing of the human itself often remains a stereotypical archetype. This is understandable because first, the characters' actual roles in games tend to be stereotypical, and second, it's not usually the concept artist's job to delineate the nuance of the human inside the combat armor. Unless you have live casting, nobody really knows who the character is until a production artist starts modeling. With this kind of setup, there isn't much of a chance to find out the bare attraction of these characters. Only a character with soul is capable of telling a good story, or delivering a message. Without real characters, the whole product winds up being an empty killing experience [although that is often all the game calls for]. Establishing the character inside the suit is essential if you want to create products that have a lasting effect on players, and on history.

can play his role as long as the body type is similar. Leonard Nimoy (from the original TV series) and Zachary Quinto (from the new *Star Trek* movie) are human beings, and our brains are capable of reading more information than just those visual language identifiers. On the surface, it's easy to tell them apart with a few words. But it's very challenging to describe the specific face shape or nuance that makes up Leonard Nimoy or Zachary Quinto's Dr. Spock (see Figure 2). It would probably require a great author to describe it in words. Likewise, it requires a great artist to describe in art. What makes a human character human is those non-visual language elements. Typical game productions don't pay much attention to that.

FINDING BACKGROUND

» If a particular game requires that players slaughter a bunch of enemies, there needs to be a reason. For example, the main character hates aliens because he saw them kill his parents. That is enough motivation to inspire the player to kill. In order to make it a proper story though, it needs one more step. For example, the player killed countless aliens in order to take revenge for his parents. However, he discovered the aliens have a good side, through a relationship with a particular alien that changed his views. The alien became his close friend. Regular video games have this level of story at best.

In order for the audience/players to really feel emotionally involved with the story, we need to go one step deeper. For example, the main character wants to say he hates

VISUAL LANGUAGE

» Concept art is full of visual language—not only costume designs like helmets, mantles, jet-packs, artificial muscles and jewelry, but also scars, tattoos, pimples, and even a five o'clock shadow. This is all common language to provoke familiar ideas. But organic objects are very complex, and especially when looking at human beings, our minds are capable of differentiating a lot more than simple visual language. The pointy ears and hairstyle of Dr. Spock from *Star Trek* are visual language tropes. That's very good costume design, and everybody remembers it. But that does not mean anybody

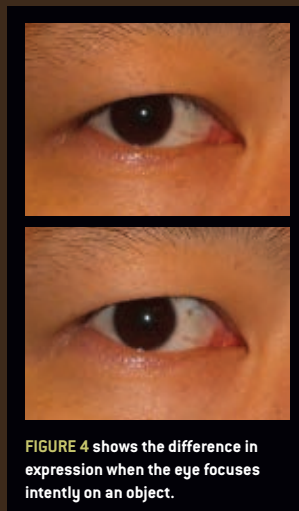


FIGURE 4 shows the difference in expression when the eye focuses intently on an object.



GD10

Game Developers Conference[®]

March 9-13, 2010 | Moscone Center, San Francisco

www.GDConf.com

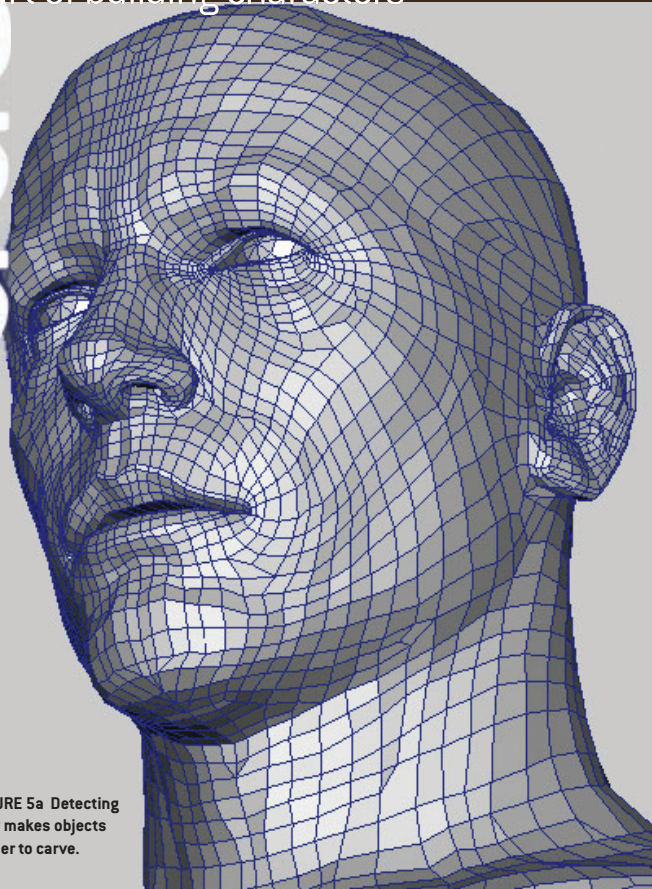


FIGURE 5a Detecting flow makes objects easier to carve.

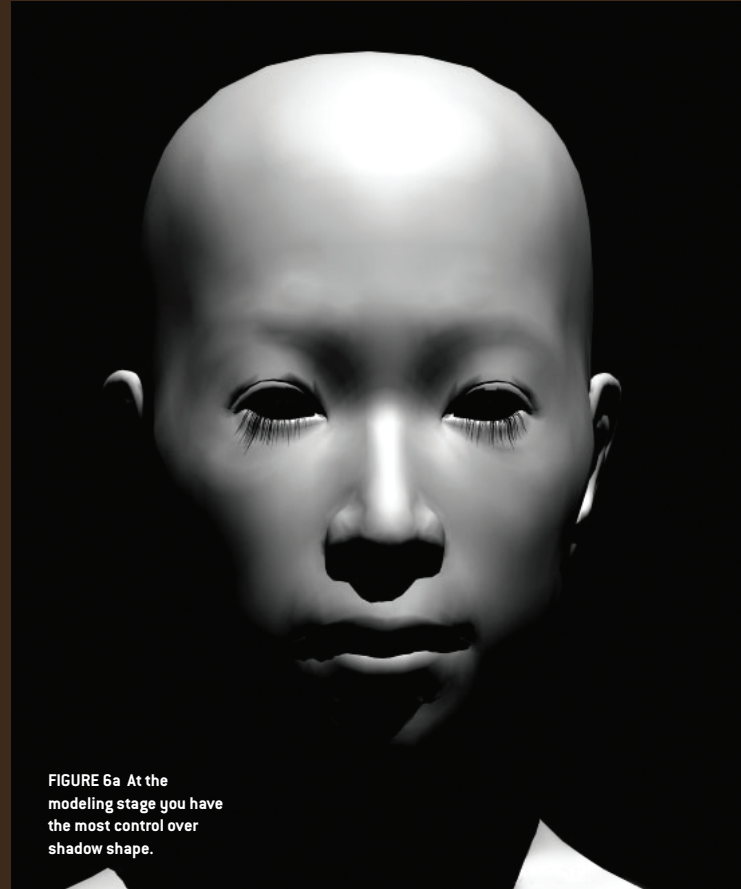


FIGURE 6a At the modeling stage you have the most control over shadow shape.

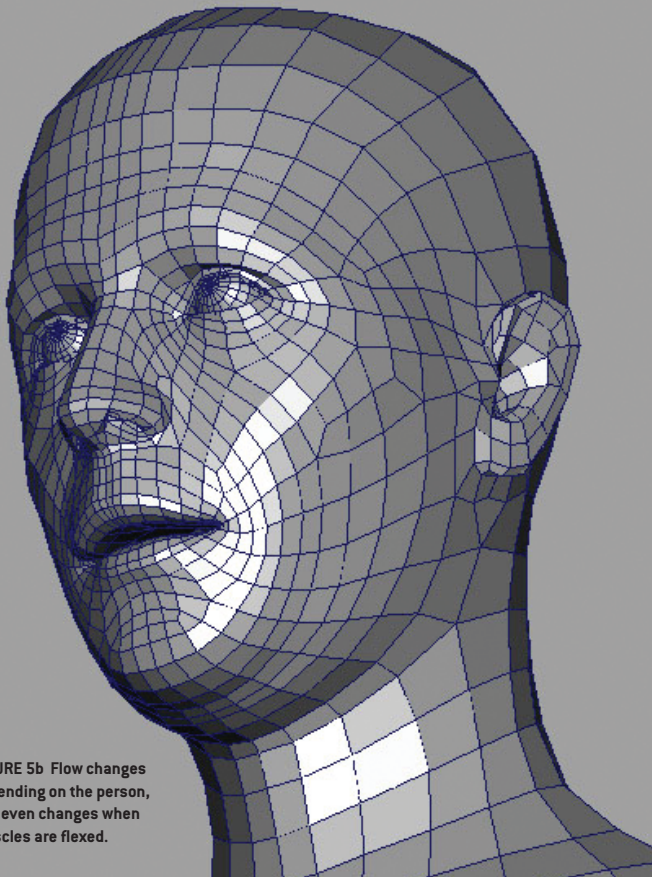


FIGURE 5b Flow changes depending on the person, and even changes when muscles are flexed.



FIGURE 6b.

aliens, because it helps him stand out among those who favor them. In truth, he's not really sure whether he hates them. One thing he is certain of is that it cheers him up when he gets attention from everybody after killing aliens, and he can play the big outlaw. He likes it. It makes it easier for him to get girls. He gets excited when he tells people his parents' sad story. That extreme delight comes to the surface, and a subtle smirk crosses his cheek. And because he tries to prevent the expression from surfacing, an odd strain appears on his face, making a nice agonizing effect. He is not always happy. He is worried that he may have to continue being this way for the rest of his life. He is tired from pretending to be someone that he is not. Deep inside, he's actually scared when he looks down the sights at an alien.

Then, a strange alien appears in front of him. It acts differently, getting attention with a divine messiah-like quality. Our hero realizes it's a good time to change his character back to who he really is. This way he can return to being a regular person without having to expose his doubts. He has saved his pride and lived happily after. This is character development. We need to understand the inner character beneath his surface persona.

FINDING FLAWS

» Finding flaws in your characters can bring them away from a false perfection, and creates great intimacy. Asymmetry is the typical method. The human face is not symmetrical—making the eyebrows unbalanced, or making one cheek sag compared to the other side, or even adding a distortion of the entire skull—these little things bring surprising intimacy. Little pores opening on the tip of a pretty woman's nose, a belt of fat that appears under the jaw line when a person looks down, or a belt of fat over her pants line when she leans over, these things add some idea of who the character is.

Yellow stained teeth? One tooth missing? One cauliflower ear? That's the typical method of adding imperfections in games, but really it's just simplistic visual language. Certainly it helps describe the role, but it does nothing to add real character depth. These sorts of tropes come from just lazily adding

random details on characters, such as pores, or wrinkles on the lips without reason. This only serves to divert the audience's eyes from the character itself to unimportant details—unimportant because those details aren't there to convey any particular piece of the story. For example, if you add realistic wrinkles on a character's lips, and it could communicate the idea that the place is pretty dry or the character is tired. This represents a failure if that is not the message that you wanted to convey. Every tiny element should be part of the final message.

FINDING SUBTLETY

» After understanding everything about your characters, the next step is determining how to reflect this character in computer graphics. Like the little smirk in the story above, you should find signs that can reflect the inner emotion.

See Figure 3 for an example of these subtleties. The image on the left is a typical mouth shape to be used in production—easy to rig and weight. The image on the right has some good subtlety that shows her personality and history. She was not happy with her small lips, and always tried to make them wider. It became natural after years of trying.

As another example (Figure 4), eyes change when a person focuses intently on an object. Probably the muscles around the eyes tighten so they can adjust the focus—there is a huge change of impression when the eyes focus, even though there's not much change in shape.

FLOW

» It's now the era of ZBrush. Especially in the game industry, details of 3D models are increasingly crafted in ZBrush, so the flow and edges of a surface can be modeled like clay, using a polygon structure. [I would not like to call this an edge loop, because the concept for edge loops deals with subdivision and tessellation, whereas polygon structure is meant for flows like this.] An edge needs to go along a contour, and since one polygon can only share four edges, it's a challenge like a puzzle. Ultimately you need to pick some edges and throw some away to maintain the quad polygons. I would say this is the most painful



FIGURE 7 Director Kon Ichikawa often lit his actors with strong cross lighting.

process and takes the most time when creating a 3D model, but this is how good models are made. Lots of time is spent on these areas that are seldom seen by non-professional eyes. ZBrush or Mudbox will release you from some of this pain, and four-sided quad polygon limitations (see Figures 5a and 5b).

With these software packages, the polygon structure does not have as large a role as it used to, but still cannot be ignored. Outside of games or in some situations like softbody simulation, you can't get away from it.

SHADOW AND SHAPE

» Normally we may think that we are designing 3D models,

but actually I think we are not. Even though we're building a three-dimensional polygonal object, the final output is always 2D. There is ultimately not much difference between painting and 3D modeling, because at the heart of it you're designing a picture, or sequence of pictures. The only difference is whether it's created with two-dimensional methods or three-dimensional ones. So how realistically do you feel you are designing that 2D output when modeling in 3D?

Shadows are the biggest factor here. We are designing more for shadows than we are for the shape. The right shadow falls on the right place if the model is right, within a

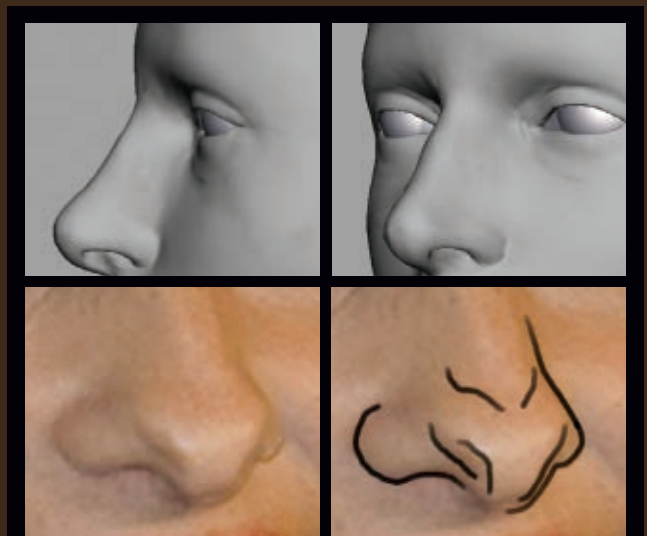


FIGURE 8 A nose that appears straight actually has a lot of complexity.

proper lighting scheme. If the resulting composition does not seem strong enough, and if the character does not appear strong enough, it's very possible that the shadow shape is wrong. You have an incredible amount of control at the modeling stage, and it's here that you can generate shadow shapes quite flexibly (see Figures 6a and 6b). If you can't hit the right shadow after hundreds of iterations, maybe the target impression that you are trying to accomplish differs fundamentally from your lighting.

As a good example of this, film director Kon Ichikawa often utilizes light coming from the side, so the faces get a clear shadow cast from their nose even with less rugged faces of Asians. Half of the face would be in shadow if he tried this trick with Caucasian actors (see Figure 7).

It would help to check the costume design of your character to see how a busy shadow runs across it, and you might also want to compare it with the complexity of the background. This will help you to establish and control the balance of shadow.

ILLUSION

» People have an amazing ability to recognize other people. You will be surprised if you try using a 3D scanner to recreate a head—the result will be different from what you expect. If you try to capture the entirety of a head with your naked eyes, even though our eyes are only three inches apart, it's difficult to do accurately. So with a scanned head, you can be sure there's going to be some mass and volume you didn't expect, which will result in an unexpected silhouette when you move your in-game camera. A straight nose when viewed from the side may be slightly hooked (see Figure 8). A cheek may actually be puffer when you see it from certain angles. Great sculpture changes its silhouette elegantly as you physically move around it, and every moment is a surprise. The concept applies to CG and you should be prepared for it. The illusion should betray your sense of volume every time the angle changes.

POSE AND ACTING

» A good character should have a good signature pose. It's easy to recognize Spider-Man by his

gymnastic flying silhouette, for instance. But again, that is visual language, which belongs in the realm of concept design. What I mean here is the posing that defines the character's personality. Good film actors change their body mannerisms for the roles they play. The neck angle and spine curve are typical targets. Simply pulling the spine in and out for default poses gives the character a really different personality.

You also rarely see a profile shot in movies, because actors rarely turn on the axis that's perpendicular to the camera. Good animation doesn't do this either, though it still happens if the director isn't good.

See the image at the start of this article for an example. The story behind this image, created for a serious game, is that this girl has HIV, and she prostitutes knowingly. A guy (we do not know who is—this could be a customer, could be police) is approaching her. I needed complex emotion on her face to indicate guilt and ignorance, but since I did not have time to add real facial expression, I solved it with simple posing. The girl successfully conveys some of the complex emotion I was looking for.

It's a fun process to decipher how your favorite actor creates his role through poses and movement, setting aside typical acting aspects like voice tone or eye movement.

FACIAL EXPRESSIONS

» Human beings live from moment to moment—we are not really conscious of what we do between describable actions. For example, when you wash your hands, your focus is on your hands. Next, you want to wipe your hands, so your focus jumps to the towel. People are not exactly aware of what they're doing while moving between these focus objects. In these in-between moments you can redirect focus to create good acting. The same idea applies to the face. A smile, a frown, anger, or even lip syncing, all are examples of visual language. Between these states, that's where you can create real character. If you step-pause a movie at home, you'll find actors making funny poses and faces between the conscious points. That's a good subject to study. Boring acting like simple smiling or frowning may turn into some attractive expression by injecting this type of essence.

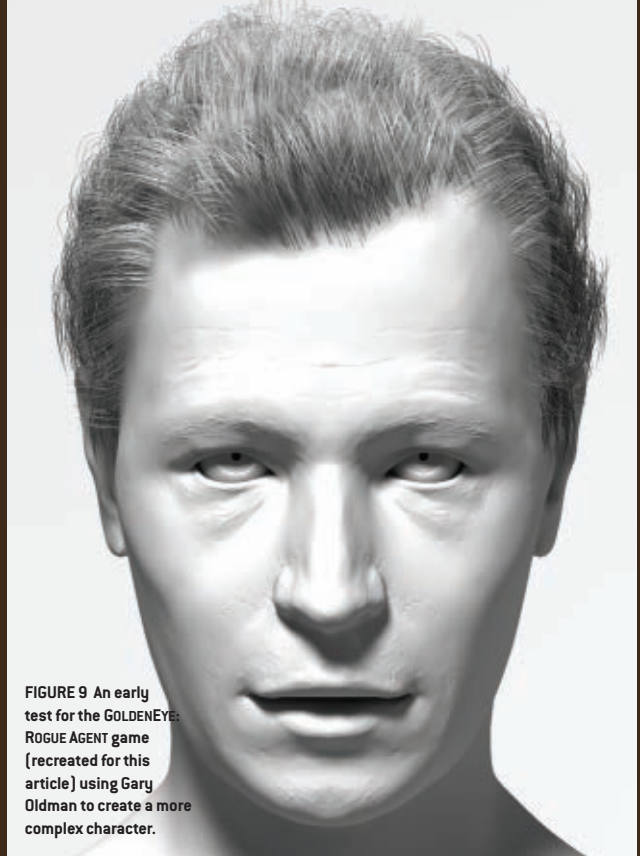


FIGURE 9 An early test for the **GOLDENEYE: ROGUE AGENT** game (recreated for this article) using Gary Oldman to create a more complex character.

DESIGNING HUMANS

» There is no guarantee that you can find a perfect cast. Even if you do, there are often many legal reasons that we cannot use the particular actors we want for our animation. That's when you have to design a human, and the result needs to be more attractive than a regular human. It must be as attractive as an actor. I think this is one of the most difficult things to accomplish. It's easier to take a specific human target and try to model it, but if you can't use the actors, you can't use them! Figure 9 shows an attempt at using a real actor (Gary Oldman) to evoke a certain mood.

The typical way to design a human is to pick stamps of features from many people and combine them like a police composite drawing. It usually takes a hell of an effort to blend them so that all parts and skull shapes work in harmony, and even then there is no guarantee that your results will look strong. You might need to go back and start choosing feature stamps again if the result isn't strong enough. Designing a human is a repeating sequence of this trial and error process. When I do this, I need to be ready to get completely exhausted, and still there is no guarantee of getting a good result.

Sometimes character descriptions are difficult to visualize. For instance, consider this one: "35 years old, but still looks somewhere in middle

her 20s, a mix of Arabian and North African. Her eyes are solemn yet very determined. Although she is not a classic beauty her beguiling serenity makes everybody mesmerized." It is cruel if this complex requirement only falls on a character artist. As CG and game fidelity goes up, more actual references are required. I think it's worth having auditions or working with domestic and oversea casting agencies to help artists when modeling.

FOR ART'S SAKE

» Various elements need to be considered before entering full production. As the technology improves and budgets go up, it's becoming very challenging to tie each component into a single direction. But great impact cannot be achieved without all elements working in harmony. Masterpieces are borne when everything works in sync. ☺

[All images created by Takayoshi Sato with the exception of Figures 2 and 7.]

TAKAYOSHI SATO is art director at *Virtual Heroes*. He also created the character designs for the recently released *FATALE*. Previously he completed all cinematics for *SILENT HILL* single-handedly, and creative directed *SILENT HILL 2* and its cinematics. He has also worked as an art director and producer at *Electronic Arts*. See his work at www.satoworks.com.



Unreal Technology News

by Mark Rein, Epic Games, Inc.

Canadian-born Mark Rein is vice president and co-founder of Epic Games based in Cary, North Carolina.

Epic's Unreal Engine 3 won Game Developer magazine's Best Engine Front Line Award for three consecutive years, and it is also the current Hall of Fame inductee.

Epic's internally developed titles include the 2006 Game of the Year "Gears of War" for Xbox 360 and PC; "Unreal Tournament 3" for PC, PlayStation 3 and Xbox 360; and "Gears of War 2" for Xbox 360.

Upcoming Epic Attended Events:

GCAP

Melbourne, Australia
December 6-8, 2009

DICE Summit

Las Vegas, NV
February 17-19, 2010

GDC 2010

San Francisco, CA
March 9-13, 2010

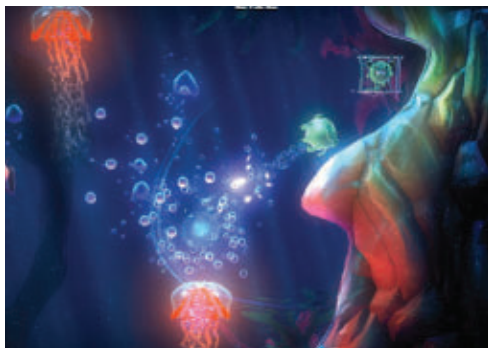
Please email:
mrein@epicgames.com
for appointments.



EPIC RELEASES UNREAL DEVELOPMENT KIT, THE FREE EDITION OF UNREAL ENGINE 3

Epic Games recently launched the Unreal Development Kit (UDK), a free edition of Unreal Engine 3 that provides community access to the company's award-winning toolset like never before.

UDK is available to anyone interested in using 3D game engine technology, including game developers, creators of 3D visualizations and simulations, hobbyists, students, researchers and digital filmmakers. Anyone can start working with UDK by downloading the latest software release at www.udk.com.



UDK-powered "Whizzle" by Psyonix Studios

Also available at the UDK web site is a developer diary, technical documentation and source code for the first project built from the ground up with UDK. "Whizzle" is a vertical scrolling puzzle game by Psyonix Studios that was developed in under two months by one artist and one programmer.

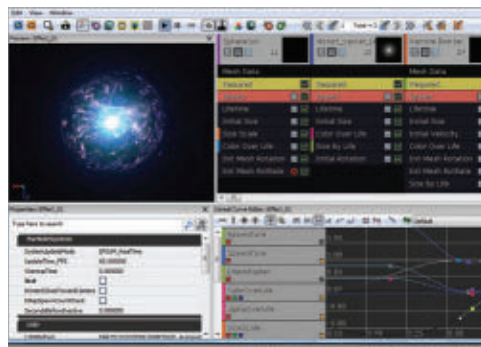
UDK provides free access to the same great tools used by many of the world's best game developers and publishers. Unreal Engine 3 is a constantly evolving game engine, and UDK contains all the most recently added features and technological enhancements, including many that have yet to be seen in an Unreal Engine game.

There is no charge for noncommercial or educational use of UDK. The software package is free for all schools and students to use, and anyone can release free UDK-powered games.

Those wishing to develop software for commercial purposes should refer to www.udk.com/licensing.

Commercial terms have been structured to make it easy for independent developers, start-up firms and seasoned professionals to use UDK with minimal financial barrier from concept to deployment.

Over 100 academic campuses use Unreal Technology as part of teaching game development-related courses, and colleges with plans to incorporate UDK into their curricula include the University of Pennsylvania, North Carolina State University and The Art Institute system of schools, with many others to be announced.



Editing particle effects in UDK

Epic is committed to supporting UDK users with high-quality documentation and ongoing, free software upgrades. Over 200 pages of newly unlocked reference material are available at the Unreal Developer Network (udn.epicgames.com), and users can access dedicated forums at www.udk.com/forums.

Over 50,000 users installed UDK within its first week of release, and Epic has shipped optimizations for stability and performance through a second beta version.

3D Buzz (www.3dbuzz.com), the leading provider of Unreal Engine 3 training, will soon release dozens of free UDK video tutorials to complement the company's existing Unreal Engine 3 video training modules. Over 100 Unreal Technology videos are viewable at the 3D Buzz web site free of charge.

"Mastering Unreal Technology, Volume I: Introduction to Level Design with Unreal Engine 3" and "Mastering Unreal Technology, Volume II: Advanced Level Design Concepts with Unreal Engine 3" are recommended for learning Unreal Engine 3. 3D Buzz authored both textbooks, published by Sams, and each come bundled with a free, downloadable copy of "Unreal Tournament 3" for PC.



For UE3 licensing inquiries email:
licensing@epicgames.com

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

WWW.EPICGAMES.COM

brütal legend

post mortem

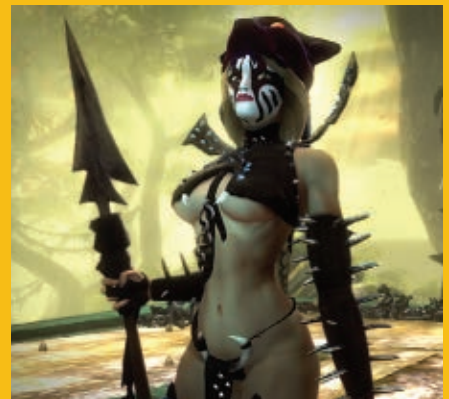
C A R O L I N E E S M U R D O C

IF THE ADAGE "GO BIG OR GO HOME" APPLIES TO ANY SOFTWARE DEVELOPMENT EFFORT, IT APPLIES TO THE making of BRÜTAL LEGEND. As we did previously with PSYCHONAUTS, Double Fine once again bet it all on innovating—this time on a game borne from the FULL THROTTLE side of Tim Schafer's mind. BRÜTAL LEGEND is a molten, balls-forward, third person, open world, strategic action-adventure interactive ride into the very soul of heavy metal.

The development story started out simply enough. After shipping PSYCHONAUTS, Double Fine created a collection of concept work, a pitch document and a game trailer intended to capture the spirit of BRÜTAL LEGEND. Most publishers we spoke to were interested in the game concept, but their questions commonly indicated that they didn't understand where we were headed with it. Questions were posed to us, such as "Why heavy metal? How about rock, or country, or hip hop instead? Why would you want to play as a roadie? How about playing as a rock god?" One publisher, Vivendi Universal Games, did not ask these questions in the pitch meetings, or in any meetings. They understood the game for what it was, and signed it for what they knew it could become.

We started development by focusing first on the multiplayer mode of the game—our thinking was that since we'd never made a multiplayer game before, figuring that out would be our top priority. It took 16 months to do so—in Rocktober 2006, we delivered a fully playable Ironheade vs. Tainted Coil skirmish to our publisher. At Vivendi's request, we then focused on the single player campaign, expanding its scope well beyond the initial design. This is the period in which we added the voices of Jack Black and a host of other celebrity talent to the game, as well as other enhancements that solidified the vision for the campaign experience that we ultimately shipped. In June 2007, we delivered the first meaningful portion of the single player game and also promptly admitted that all of the changes we had made to the game content put us way behind schedule. Our first schedule revision extended the project by ten months, the second by another seven. Initially scheduled to be released in May 2008 under the Vivendi Universal Games/Sierra banner, BRÜTAL LEGEND finally shipped on October 13, 2009, published by Electronic Arts.

Double Fine encourages innovation, but that drive also means we can't always rely on previous experience to predict how a feature or an approach will turn out. On BRÜTAL LEGEND, the practice of continuous iteration and concept refinement led to a number of prototyped ideas, many of which survived to ship in the final game, but just as many of which were left to digitally fossilize in the annals of Perforce. Whittling down to ten the list of things that went right and wrong during the development of BRÜTAL LEGEND presented a considerable challenge. Here are some lessons that were the most surprising or impactful.





brütal

WHAT WENT RIGHT

1) PUSHING CREATIVE LIMITS. BRÜTAL LEGEND was to be the interactive amalgamation of the over-the-top ridiculous (yet deadly serious) world of heavy metal. We were reverent fans of the genre and felt it would be an honor to bring that world to life. BRÜTAL LEGEND began with a simple list—a game that embodied everything that could be found on a heavy metal album cover: chrome rivers, pools of blood, volcanoes, caves, fire-breathing metal beasts, laser panthers, bladehenge and beerhenge, dominatrices, latex and chains, disembodied undead heads ...

To that we added the core gameplay. We wanted to make a brawler adventure game, where the player was a heavy metal roadie who evolves into a rock god over the course of the game. BRÜTAL LEGEND on the Xbox 360 and the PlayStation 3 would have the brawn of an action game and the elegance of an RTS.


We learned early on in our relationship with Vivendi that RTS was a naughty word in the console space, so we stopped calling it by that name and, by extension, so did Electronic Arts—positioning the game largely as an action title in the marketplace. We wanted our RTS to exploit the consoles' advantage; putting the player in the center of the action. We wanted to give the player intuitive control of a character that could perform a variety of badass movements and abilities and also allow the player to personally bond with that character. And we wanted that character to command dozens of masterfully-dialogued troops simultaneously.

One of our biggest challenges in solving the RTS accessibility issue was squad orders. It took numerous attempts and countless focus tests, but we ultimately decided on a simple unification of the orders interface, wherein the AI behaved as the player would want them to on the battlefield. The player would have four orders: 1) "follow," where Eddie gives his troops commands to move to a specific location and possibly attack, allowing the order to be given only within "shouting range," a relatively large distance around Eddie that did not encompass the entire map—solving the forced (ignore path enemies/obstacles) and non-forced

(engage enemies along the way) issue by making the "follow" order non-forced when Eddie was near his troops and forced when Eddie was far away, 2) "defend," where Eddie could command his army to stop moving and hold position, aligning them in the most advantageous manner (melee up front, ranged behind, support in the rear) facing the camera, 3) "move," where the migration would be forced until the army is close to its destination, at which point it would engage nearby foes, and 4) "charge," a non-forced move to the enemy that is closest to the average position of all nearby enemies if your army was not attacking, and a forced move to the attack position even if that meant disengaging from their current activity if your army was attacking.

The Double Fine incarnation of a console RTS occurred to us not in an early pre-production meeting, but over years of painstaking iteration and reinvention and rework. We tested our progress in periodic "Mandatory Hour of Fun" sessions, where the entire team played the latest build and then met as a group to discuss what was working well and what was frustrating or could be made better. This open forum for the exchange of ideas as well as the momentum for continuous iteration fueled profound changes to the core game mechanic over the course of development.

2) SCRUM. Prior to starting work on BRÜTAL LEGEND, the Double Fine team had spent the previous 5 years developing PSYCHONAUTS—the last two years of which consisted of a giant, grueling crunch wherein the company lost its initial publisher and nearly shut its doors before ultimately releasing the game. When the euphoria of having shipped our first title wore off, it was apparent to all of us that Double Fine did not develop games the way other studios did, and that a different system of product development needed to be put in place. The main cause of PSYCHONAUTS' horrifying crunch was due to our continued development of the game features even after the levels were built. With each improvement to the game mechanics came a corresponding rework of all of the levels. Lather, rinse, repeat. Double Fine, and notably Tim, needed to play



GAME DATA

DEVELOPER Double Fine Productions, Inc.

PUBLISHER Electronic Arts

PLATFORMS Xbox 360, PlayStation 3

RELEASE DATE Rocktober 13, 2009

NUMBER OF IN-HOUSE DEVELOPERS 74

NUMBER OF EXTERNAL CONTRACTORS 9

BUDGET \$24 million

DEVELOPMENT TIME 4.5 Years

TYPICAL WORKSTATION Dell Quad Core Xeon, 4GB RAM, 150GB, NVidia GeForce7800, Windows XP

SOFTWARE USED MS Visual Studio 2005, Autodesk Maya 2008, ZBrush, Adobe Photoshop, Adobe After Effects, ProTools, Sound Forge, Perforce, Microsoft Xbox360 XDK, Sony PlayStation 3 SDK, Internal suite of development tools

TOTAL LINES OF CODE
Game [C++, excluding middleware]: 554,736
Game [Lua]: 42,745
Tools [C++ & C#]: 52,263
Tools [MEL]: 85,294
Tools [Python]: 34,079

TOTAL NUMBER OF PERFORCE CHECKINS
164,863

the game, live it, breathe it, let it steep over time, and iterate continuously on what makes the game fun and funny.

After research into methodologies, we were drawn to the advantages of agile software development and decided to adopt Scrum. Within the first few months of BRÜTAL LEGEND development, the team was practicing Scrum, and the initial payoffs were impressive. Scrum's emphasis on features over systems, on rapid prototyping and iteration, on cross-disciplinary teams, on people over process, and on the creation of a potentially shippable piece of software every sprint/milestone made the game playable at a very early stage in development: by month one we

had a renderer, terrain, and a playable character (Eddie Riggs), by month two Eddie could drive his hot rod (the Druid Plow) around the terrain, and by month three Eddie could run over endless numbers of headbangers with his Druid Plow around a terrain height field. Hilarity ensued.

We applied Scrum not only to meta-game creation, but to micro-projects as well. At the very start of the development process, we had no idea how to make an RTS, and had no suitable engine with which to make one. We solved both problems by creating prototypes with an off-the-shelf PC engine and with which a number of our team members had some familiarity—Unreal 2.5. The design demands of BRÜTAL LEGEND were such that trying to develop the game using an existing FPS engine would have proven difficult, but having the initial access to the flexibility of Unreal Script meant we could test some of our early RTS ideas right on our development PCs. This approach allowed our designers and gameplay programmers to be immensely productive right away, while the programming team went to work building our new engine. This very early glimpse at the design challenges we would face during development, and the opportunity to iterate on something quickly with Unreal Script, gave us invaluable direction into how to architect our new engine and critical insight into the mechanics that would come to define BRÜTAL LEGEND.

There was a notable downside to Scrum that bears mention in spite of our success with it. Our implementation of Scrum encouraged a pre-production mindset far too long into production. Scrum neither encouraged defensive programming, nor the practice of designing systems that scaled well. We took a number of systems to 80 percent, enough to prove an idea, or extend and refine it in a future deliverable. But we found in the last few months of development that the remaining 20 percent was another 80 percent of the effort—leading to some unexpectedly crunchy milestones for the hardest-hit team members. Once the project was in production, because of the reliability and lower risk of art creation, a waterfall approach was more optimal than Scrum.



Even with these caveats, Scrum allowed the team to quickly test wild theories and not only keep the best ones, but also spit-shine them with continuous iteration over the entire course of development. It was a significant process improvement for the studio.

3) BERT'S BOTS. Double Fine employs a full-time testing army of two. We relied on our two testers to smoke the daily builds of the game, as well as to pound on any new features we intended to get into the next deliverable. It was obvious very early on that two testers alone could not keep up with the stability demands of the game, nor was it financially possible to hire a full time test staff for the duration of the project. Our solution was to develop an automated testing system, which we affectionately named RoBERT, after Bert, our software test engineer.

The automated tests began as an experiment, and the initial system took about a month to put in place. The first simple scripts summoned every character and object in the game. They proved immediately useful in finding warnings and crashes. Running the tests on a regular basis allowed these crashes to be found in a timely manner, narrowing down the causes of the crashes to the

more recent changes made to the code base. Programmers started running automated tests locally so that they could test risky code changes before checking them in. Tests would also be run to help reproduce a bug.

One particularly crafty programmer came up with the idea to borrow idle Xbox 360s and PlayStation 3s to run automated tests. Team members could always end a test and take their machine back, but it was useful and efficient to use idle machines to run tests 24/7. We estimate that automated tests in the bot farm ran for a combined total of 147,000 hours.

The automated test system was so successful that we extended it to include tests with two armies battling it out and balance tests to determine how powerful each unit was. Bot functionality was then added where input could be simulated. This allowed a test to perform moves exactly as a player would in an actual game. Bots were used to perform attack combos, and a variety of other moves with every squad, as well as to find stuck-spots in the world. They were also used in multiplayer tests, which were invaluable in finding desyncs in our peer-to-peer lockstep networking system. As long as there were available machines, 1v1 and team multiplayer tests could run on a

brütal



regular basis. Finally, the bot system was expanded to play through the campaign. Test settings were added for playing through the secondary missions, exercising the Motorforge, upgrading equipment, or failing each mission before completion. Memory reporting was soon added to the campaign tests so programmers could track memory usage and leaks.

Automated tests could be run remotely and crash reports emailed to interested and responsible parties. Doing so significantly simplified running multiple tests simultaneously and tests were often run using differently-configured builds. Early on, most tests ran using a debug build so programmers could attach to a crashed machine and more easily debug the problem. Late in the project, tests ran a special release build to find release-only crashes. There were some limitations to the system, such as a crashed machine in the bot farm being rebooted by the machine's owner before a programmer could debug the extant crash. And the bot never learned to path or drive, nor could tests be run in the pause or front end menus. We will be making these improvements and additional expansions to RoBERT for future projects.

4) TOOLS SUPERPOWERS.

Double Fine wrote a number of tools that, when added together, made for a

streamlined development pipeline. It's hard to single out any one tool and express how it enabled efficient development. A couple of our weapons are described here.

The MUE (Multi-User-Editor) is our collaborative open world building tool which forms the backbone of our world production process. Its primary function is to allow designers and artists to work simultaneously on our large world. The BRÜTAL LEGEND world is divided into different "tiles", and the MUE allows for individual tiles to be checked out and edited by different individuals. In addition, any or all of the thousands of entities populating the world can be checked out and edited by different individuals.

The MUE also offers a proprietary set of tools for editing height map terrain—a bit like ZBrush lite, but specifically optimized for editing height map terrain. In addition to the expected suite of sculpting and smoothing tools, the MUE has an innovative sculpting tool known as "terrain stamping" which allows artists to import pieces of mesh, arrange them on the terrain and then click a button to "stamp" that shape into the height map terrain. It was an especially useful technique when creating critical gameplay areas that had specific height, angle, or width requirements for the terrain. In addition to sculpting the terrain, the MUE allows artists to paint down blended terrain materials and

ambient mesh (grass, bushes, and so forth) in real time.

The MUE runs inside a Maya window, but under the hood lies a powerful SQL database. While having a database back-end may not sound exciting, it served as a powerful safeguard against data loss and enabled rapid worldwide changes. If an artist wanted to change all the poster trees into tire trees, he would run a one-line search-and-replace python script, which also handled the checking in and out of the data.

A second tool (called the Rigerator) was created to support our rigging and animation tools pipeline. Faced with the daunting task of creating and animating around 150 unique characters, some with upward of 40 facial shapes, as well as around 300 cinematic scenes, it was critical to automate the character rigging and animation pipeline as much as possible. Due to the high volume of art revisions made internally, we also had to enable making changes easily. To achieve this we started by using Maya referencing and naming conventions assigned automatically by the tools to minimize user input, minimizing inconsistencies between users and files. To author and edit character rigs, we created an automated process wherein a user could match a simple skeletal layout to their character's proportions and simply press a button to create the complex animation controls instantly with proper placement and naming based on whatever unique size or shape that character had. Prior to the development

of this tool, the process would take one to two days to complete manually, but animators were now able to rig their characters instantaneously.

Additionally, since unique characters require many unique animations, we developed a tool that managed a library in which a character's unique poses and animations could be easily saved and loaded onto other characters in the game. While speeding up the animation process, this Anim Toolbox also kept all character animation sets creatively cohesive regardless of author, and it applied to both in-game animations as well as the cinematic scenes for the game. Without enabling this workflow we could not have created the vast amount of animation in the necessary time.

5) BIG NAMES. Few games, if any, can boast the number of licensed music tracks or the number of signed celebrity actors/musicians that BRÜTAL LEGEND can. It was a monstrous and painstaking endeavor to select each of the artists and tracks to include in the game, but it was worth the effort.

For licensed music, Tim and music director Emily Ridgway chose songs that had substance and credibility. Each song needed to both fit its use in the game and also be embraced by metal fans. For BRÜTAL LEGEND to truly honor the genre, it had to be rich with heavy metal presence, since the music, lyrics, and imagery served as the game's creative direction. Once songs were selected, it was then a Herculean effort to seek out, negotiate, budget,



sign, and shepherd through the licensing process over 100 music tracks, each with a master and a synch agreement, many across multiple rights holders, and some from bands that had long since split up. We sought permission for each song individually, and in some cases, Double Fine's personal outreach to bands on behalf of BRUTAL LEGEND resulted in the creation of original music for the game by the artist, such as Lita Ford's "Betrayal." The final soundtrack to BRUTAL LEGEND is epic, and the oft-unheralded licensing process that took the excitement of identifying and selecting tracks and translated it into permission to use is something we're proud of.

While our music content dreams were lofty from the start, we had not initially planned to cast Jack Black as the voice of Eddie Riggs. We approached him with the prospect after learning he was a fan of PSYCHONAUTS, and, as a metal fan himself, he signed on to do the voice acting for the game. In addition to Jack Black, a number of other high profile musicians were identified and sought to provide the voices for the characters that would personify them in the game and honor their contributions to heavy metal. Each required outreach, negotiation, scheduling, and directing. Personality and humor always define the characters in a Double Fine production, and BRUTAL LEGEND is another example of casting and voice direction done right—except this

time around, the cast is a little more, well, famous.

Surprisingly for us, the beefy licensed music soundtrack, and the addition of a celebrity cast to voice the characters, especially the voice of Jack Black as Eddie, helped to inspire Tim's writing, motivate the team, and excite publishers.

WHAT WENT WRONG

1} MIDDLEWARE. The prospect of building new next-generation tech on multiple consoles at the start of the BRUTAL LEGEND development cycle was a daunting endeavor for us to consider, and neither the practice of Scrum nor our own ambition would allow us to write every system necessary for the game.

We recognized the advantages of middleware use early in the game's development. Selecting the right middleware to integrate allowed us to get the game off the ground early and facilitated the rapid testing of new ideas. This advantage allowed us to make strong demos at the start of our development cycle, which helped communicate our game concepts to potential publishers when pitching the game, and also to present strong milestones during pre-production when publisher expectations for deliverables were more forgiving. Double Fine used middleware for audio (FMOD), physics (Havok), user interface (Scaleform), lip sync (Annosoft—tools and offline), and video (Bink). Though there were some false starts with other middleware packages, the decision

to distribute the development risk across commercially available software packages was sound, and the call to integrate those packages early in development paid dividends throughout the development life cycle.

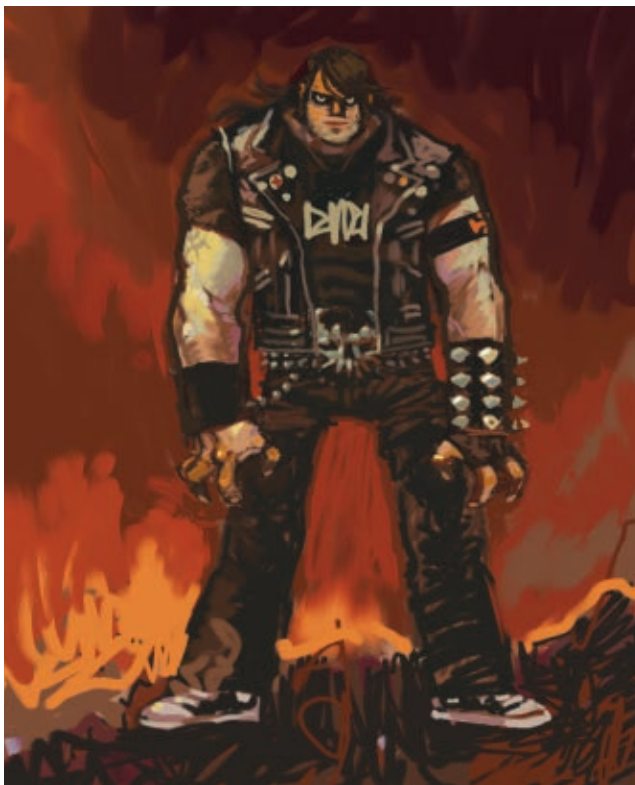
As beneficial as it was to rely on commercial software for various systems, Double Fine's batting average for selecting appropriate middleware was .500. We ended up selecting and integrating a number of middleware software packages, and then later discarding them because they no longer served our needs as the project evolved. The middleware with which we ultimately shipped still demanded a tremendous investment in its integration, requiring far more staff than we had to handle the emerging stability and performance issues. We were informed by multiple middleware companies that we were uncovering bugs and failures in the software that had not been found previously due to our placing new demands on their systems. That news was both flattering and unnerving, and ultimately costly to manage during stressful development periods. It was a tremendous relief to have the kind of tech support that we did from the individual middleware companies—without their timely attention and desire to collaborate on our title, we would not have shipped the game at all.

While we already know we plan to continue to use certain middleware packages in future projects, we will be committing to the use of others much later on in the development cycle or writing our own software

that specifically meets our needs. The wasted time spent integrating inappropriate or unusable middleware into our code base, then reintegrating others or rolling out our own solutions, absorbed a tremendous amount of engineering time and effort, as did supporting and debugging the middleware we ultimately did elect to use through BRUTAL LEGEND's release.

2} 11TH HOUR TOOLS. From the start of development, we recognized that good tools needed to be a priority to ensure faster iteration times for our team. A mix of legacy team structure, intrinsic complexity, insufficient input from content creation leadership, poor planning, and bad luck all contributed to a significant delay in the delivery of several critical tools to the team—delays which jeopardized milestone deliveries, reduced artist productivity, and required unplanned emergency intervention at the expense of other engineering efforts. Unfortunately, we seriously underestimated what it would take to get from our brightly colored Xbox PSYCHONAUTS graphics to the rich, highly-detailed lushness of our Xbox 360 and PlayStation 3 BRUTAL LEGEND graphics.

First, in an attempt to address the lack of dedicated tools programmers we experienced on PSYCHONAUTS, we created a tools group within BRUTAL LEGEND's engineering team, and staffed it with four engineers. While this was an improvement over PSYCHONAUTS, the problem with this approach was that it facilitated the separation of tools work from the rest of the engineering



effort, isolating the accessibility of the tools to the user from the process of delivering features for the game. Often, this separation created inefficiencies in implementation, or a mismatch between the tools design and the runtime feature it supported—in several cases, this disconnect even led to sprints where a runtime team would complete an artist-facing feature without any tools support for it at all, making that feature (often a milestone deliverable to demonstrate) unusable.

A second critical mistake involved effective tools prioritization. Without strong guidance from content creation, the tools team focused initially on back end services such as tools deployment and build infrastructure, with the intent that these services would serve as a productivity multiplier on future tools development. Unfortunately, without any tools to distribute or code to build, these initial investments significantly delayed the development of content/game facing tools. Later on, we struggled to find the right balance between meeting short term needs and building tools that could scale to a full production environment. A prime example of the effect of this ineffective prioritization was the one

year deployment delay of our MUE (Multi-User-Editor). An MUE would be a massive undertaking under any circumstances, but in our case the effort was made even more difficult by a late start and a rocky path from the production and dirty short-term to the production-ready real version of the tool.

Then there was some bad luck. We decided to be an early adopter of a COLLADA-based Maya exporter pipeline. Initially, we underestimated how long it would take to implement a COLLADA 1.3-compliant parser. Then, when COLLADA 1.4 required a massive change to our schema, we had to significantly rewrite our parser. Once it was finally functional, we were faced with ongoing maintenance or unexpected limitations, and ultimately failed to recognize any of the expected upsides, like easy interoperability or enhanced leverage of a tool or pipeline utility. On a lighter note, as a result the word "COLLADA" became a swear word and elicitor of groans at Double Fine Scrum meetings. In retrospect, at least it provided us with humor.

In hindsight, we should have created a programming team structure that enabled the creation of good tools alongside the development of runtime

features, perhaps even allocating a larger percentage of the engineering team to tools early in development. Moving forward, we intend to build our tools starting with the most client-facing work first, then iteratively improve them to meet the needs of the content creators. We also intend to de-emphasize technology investments that promise to yield large theoretical future benefits in favor of investments that speak to present needs.

3} CONTENT AVALANCHE. BRÜTAL

LEGEND is not a small game. Fortunately, we thought we knew what we were facing and invested heavily in data/build infrastructure. What went horribly awry was that we both underestimated the total content push and, more importantly, didn't anticipate the huge content spike at the very end of production. From the start of the game through the end of 2008, both our rate of data churn and data growth were fairly steady and corresponded roughly to increases in staffing and team productivity. This was expected and planned and supported by the technology.

But then, in January 2009, everything exploded. All at once. After three years of development we had accumulated about 2.5 GB of optimized/packed game data. Less than four months later, we'd jumped to over 9 GB.

The central cause of this was a very large increase in asset delivery from a number of teams simultaneously. For example, we went from 0 localized files to about 100,000 in a matter of weeks. We received the high resolution video assets for the Jack Black intro and all our main menus in one heap. We made a late decision to contract additional audio work, and new ambiences and sound effects were quickly added to the game. And so on.

This simultaneous significant increase across a number of types of content put a massive burden on our entire infrastructure, in particular our build machine, Perforce server, and network backbone. To exacerbate matters, we started to see cascade effects—where a massive hit to one system (such as a check in of 10,000 .wav files) would bog down Perforce, causing a bottleneck in all of the dependent systems (like our build

server and individual check ins) and these bottlenecks would then cause other bottlenecks. It's a credit to our pipeline and build infrastructure that things never failed, but we experienced a number of severe performance degradations, many of which required emergency interventions from the engineering team. This unexpected fire fighting caused lost productivity, invariably at the worst possible times (like during preparation for a demo or a milestone).

These large content dumps also put significant strain on our runtime systems. The per-line memory overhead in the voice system was not prepared to handle tens of thousands of lines, causing us to panic about our ability to even fit on a dual layer DVD. Across the board, these unexpected increases in content caused ripple effects throughout our IO, memory, and processing profile. And because the rate of increase was both high and unexpected, the engineers responsible for wrangling these systems were pulled from their assigned work and redirected to emergency fire fighting.

Moving forward, we will be much more cognizant about working with content creators to proactively estimate the total amount of data that they plan to create and to factor these numbers into our technical designs to ensure that we meet the final needs of the product. Additionally, we plan to invest more in scalable data infrastructure in the hopes that we can be better positioned to bring new capacity online quickly should it prove necessary. With those improvements and a little luck, hopefully content avalanche handling will be something we brag about in our future projects.

4} FACILITATORS V. IMPLEMENTERS. AS

BRÜTAL LEGEND moved into production, it became clear that the team was understaffed in key implementation positions. To meet this need, we reallocated positions set aside for facilitation hires (such as design and production) to staff up more implementers (such as animators and programmers). While it was necessary to staff up on implementers, we failed to recognize until we were deep into production that our overall efficiency was reduced by this personnel trade, due to the statistical increase in designs that had to be reworked, an up-

trend in management oversights, and general miscommunication.

The price of understaffing the design department meant that we were often implementing ideas that had only been loosely discussed, before the feature had been fully planned. Serious flaws were sometimes discovered only after a feature had been partially or fully implemented. This also meant that there was always a hefty backlog of decisions and specifications, even before adding in the rework required after false starts. Since it is often at the top of the dependency chain, bottlenecking design had a deleterious impact on the rest of the team, especially considering our broad game scope.

The same understaffing symptoms could be found in the production department. Producers are responsible for communication, for resolving dependencies, for ensuring the team works reasonable hours, and for keeping the project on schedule. They ensure that the road ahead is paved well ahead of the team's arrival. With insufficient staffing, some of the production responsibilities fell to the leads, or, absent that safety net, inefficiencies were handled with overtime and stress.

In retrospect, staffing facilitation and implementation positions should have been equally prioritized. Scoping the game to meet our capacity to create it would have almost certainly meant a smaller game, but one more likely more polished and smoothly managed.

5) DOUBLE FINE GETS SERVED. In June 2009, Activision Entertainment Holdings, Inc. filed suit against Double Fine, claiming breach of contract and seeking a preliminary injunction to stop the release of the game by Electronic Arts on Rocktober 13. Less than 2 months later, the case settled out of court. I can't talk about any of that in this article, or any article really. I bring up getting sued as something that went wrong because of the impact the between-publishers transition and subsequent lawsuit had on the development of BRÜTAL LEGEND.

Let's go back a little bit. We had been working collaboratively and successfully with various groups at Vivendi for two years until Vivendi merged with Activision and we lost touch with both publishers while a lawsuit percolated. The merger

announcement and subsequent diminution in publisher contact with Vivendi personnel, especially after such a previously harmonious relationship, caused internal unrest and morale dips among the team. Company meetings often included frustrating discussions about what little we knew about the current situation at our publisher, and what the various possible outcomes would mean for Double Fine.

This demoralizing uncertainty lingered for months, during which time the leads continued to motivate the team to hit their scheduled milestones while watching our coffers run dry in the absence of any publisher payments. We learned Activision was not going to be publishing BRÜTAL LEGEND through an official press announcement issued by Activision that listed the games they would be shipping, ours conspicuously absent. Again, the team was abuzz with anxiety—and the official hunt for a new publisher began, distracting Tim, myself, and various team leads during an already intense development period.


Even after the game was re-signed with Electronic Arts, we enjoyed only a brief reprieve before the legal communications began among Double Fine and Activision and Electronic Arts. Most of the team was shielded from the drama that unfolded between December 2008 when Electronic Arts announced that they had picked up

the game for publication and July 2009 when the lawsuit settled. But Double Fine's leadership was not, and the distraction and stress took its toll on individuals and on our deliverables. The lawsuit was filed just as the game went Alpha, with a stipulation that it be heard prior to Gold Master being submitted—relegating Tim and myself and a cadre of team leaders to the unenviable job of information gathering, declaration writing, lawsuit reading, witness interviewing and all around non-game-making during the crunchiest, most critical time of development. The lawsuit took its toll on the team, on the company, on our product and on our optimism. Wrong, any way you slice it.

DOUBLE FINE, FOR METAL:

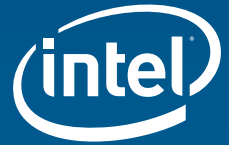
» It is both stirring and humbling to contemplate all of the epicness that BRÜTAL LEGEND coalesces in one place. Everything in the game—from the lore, to the locations, to the life, to the linguistics—has been individually handcrafted by a Double Fine. 107 licensed heavy metal tracks from 75 different bands and countless heavy metal album covers inspire the game's creative direction. The voices of celebrities Jack Black, Rob Halford, Lemmy Kilmister, Lita Ford, Ozzy Osbourne, Tim Curry, Kyle Gass, David Cross, Brian Posehn, and Wil Wheaton alongside the stellar voice acting work of veteran video game actors such as Kath Soucie, Zach Hanks, and Jennifer Hale bring the more than 150 original

characters to life through 40,000 lines of dialog that were written, recorded, edited, and hand-integrated into the game. Glenn Tipton and K.K. Downing personally wrote and recorded Eddie's, Ophelia's, and Doviculus' original guitar riffs. It takes the following words—third-person melee, single player, multiplayer, open world, driving, RTS, adventure—to describe how Eddie uses one axe, one guitar, and one hot rod to harness the power of metal to command armies and defeat evil.

In the end, 83 developers, propped up by the patience and tolerance of their families and friends as well as by the profound support of Electronic Arts, exemplified the utmost devotion to their craft and a fierce tenacity in the face of uncertainty and disappointment—surmounting publisher plate-shifting, turbulent contract negotiations, lawsuits, misdirection and redirection, focus test surprises, and extreme excesses of ambition. Armed now with a proprietary game engine, a robust tools pipeline, a talented and experienced staff, and the creative freedom and corporate mandate to innovate, Double Fine is well positioned to set forth on its next epic journey. 

CAROLINE ESMURDOC is the executive producer on BRÜTAL LEGEND and the COO of Double Fine Productions. Prior to joining Double Fine, she founded Circus Freak Studios and produced games at Accolade, Infogrames, Atari, and Crystal Dynamics. Email her at caroline@esmurdod.com.





Pay No Attention to That Man Behind the Curtain

It's easy to take for granted the video game coding and effects that bring near cinema-quality visuals to the screen—the flash of sunlight on a polished sword blade, the shifting formations of clouds moving across the sky as shadows trail them on the ground, the dark glint of malice in a character's eyes as a battle begins, the bird's eye view of an intricate forest landscape passing swiftly below. Behind these familiar and convincing graphics a host of visual computing technologies combine to build the illusions of fantastic worlds—physics to model the formation of towering cumulonimbus, game engines to control the behaviors of entire civilizations teeming with characters and mythical beasts, lighting systems to portray the subtleties of sun and fog and shadow, artificial intelligence to turn a predator's actions into a deathly drama.

Forgetting the admonition from one of the world's most popular cinema masterpieces from the 30s, we'd like to invite you to take a peek behind the curtain at the innovative, breakthrough middleware technologies being delivered by independent software vendors at the top of their game—so that the games you build can topple expectations.

Read on to learn more about middleware technologies from Epic Games, Umbra, Geomerics, Interactive Data Visualization, Simul Software, Trinigy. Through technology enabling from Intel and hard work from fellow travelers, middleware innovation is producing an exciting new era of visual computing excellence.



Visual Adrenaline—the gateway to Intel's Visual Computing software enabling, technology, and marketing resources—supports a thriving community of developers, educators, programming experts, engineers, and technologists.

For incisive viewpoints on visual computing advances, sign up for Intel® Visual Computing Dispatch:
www.intelsoftwaregraphics.com/welcome.htm.



Intel does not make any representations or warranties whatsoever regarding quality, reliability, functionality, or compatibility of third-party vendors and their devices. All products, dates, and plans are based on current expectations and subject to change without notice. Intel, Intel Core, VTune, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others. | Copyright © 2009. Intel Corporation. All rights reserved.



Epic Unreal Engine 3

Pop the hood on many of the most electrifying, visually stimulating video games on the market and you're likely to find Unreal* Technology.

The platform and tools offered by Epic provide a comprehensive development environment for bringing 3D projects to life. The Epic Games Unreal* Engine 3 supports advanced technologies from many of the top middleware developers in the industry, and the Integrated Partners Program from Epic helps bring together components for streamlined cross-platform development.

Versatility is the strong suit of Unreal Engine 3. The kinds of games created using the cross-platform capabilities of this game engine include: action (*Batman*: Arkham Asylum*; side-scrolling (*Shadow Complex*: Chair Entertainment*); MMORPG (*APB*: Realtime Worlds*); racing (*Vin Diesel Wheelman*: Ubisoft*); and role-playing games (*Lost Odyssey*: Microsoft Game Studios*). Unreal Engine 3 has also found a home in numerous 3D applications other than games, including disaster preparedness simulations, architectural walk-throughs, and training applications.

Collaboration between Intel and Epic goes back more than 10 years to when the two companies pioneered landmark-setting uses of MMX™ technology. Today, performance improvements through multi-threading are a key focus. "Intel® Threading Building Blocks (Intel® TBB) 2.1 allowed us to easily remove allocation-related multi-threading bottlenecks on 64-bit operating systems," said Michael Capps, president of Epic Games. "We are pleased to share the benefits of Intel's memory allocator with Unreal Engine 3 licensees, as well." Through Intel Software Development Products recently available via the Epic Integrated Partner Program, Unreal Engine Licensees have access to additional resources for multi-threading and performance optimization of their games.

Success stories, news, and licensing information appear on the Unreal Technology site: <http://www.unrealtechnology.com/>.





Umbra Occlusion Booster

Gamers expect Triple-A games to burst off the screen in photorealistic splendor, but the trade-offs between scene complexity and game performance present an ongoing challenge to developers.

The solution is middleware technology to optimize rendering performance. Umbra Occlusion Booster* is an out-of-the-box, multi-threaded rendering optimization library that's proven to work with multiple released Xbox 360, PS3, and PC titles. Umbra Occlusion Booster helps render complex scenes more efficiently by automatically culling invisible objects before they reach the rendering pipeline. As a result, the game developer will be able to create richer and more vibrant games scenes without compromising performance or the development budget. Umbra Occlusion Booster can also handle runtime changes in game scenes effectively. If you need to blow a hole in the side of a building or let players design custom content in your game, Occlusion Booster is there to help.

"Umbra Occlusion Booster uses highly optimized algorithms and data structures to rapidly determine what to render in each frame," said Teppo Soininen, developer relations

manager at Umbra Software. "We utilize every resource we can to let developers create larger, more complex game environments, and Intel® hardware and expertise is invaluable to us."

The collaboration with Intel includes the use of Intel® VTune™ Performance Analyzer to eliminate bottlenecks in code, Intel® Graphics Performance Analyzers to enhance code for systems with Intel® Graphics chipsets, and Intel® Thread Profiler to locate and analyze threading performance issues. To keep pace with hardware advances, Umbra uses next-generation Intel® development platforms for their software projects.

For the latest advances in rendering optimization middleware, visit www.umbrasoftware.com.



IMAGE COURTESY MASS EFFECT™



IMAGE COURTESY DRAGON AGE: ORIGINS™

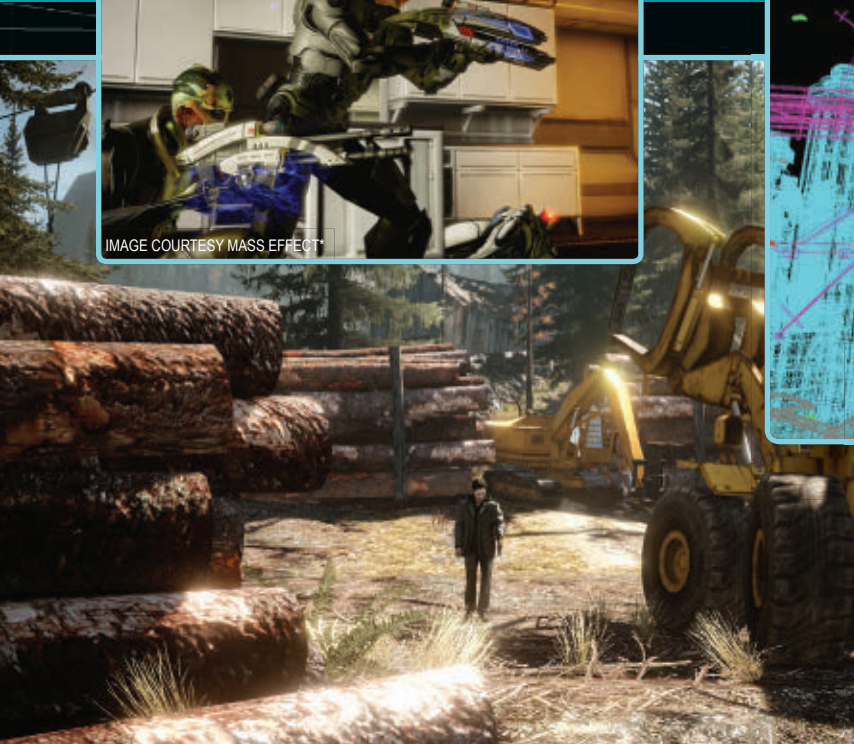


IMAGE COURTESY ALAN WAKE™



Geomerics Enlighten

Enlighten* from Geomerics gives artists the ability to interactively experiment with dynamic lighting during game development.

Delivering real-time radiosity lighting in a way that lets artists light scenes with the precision of a film director, the Enlighten software development kit provides an effective means to enhance the visual appearance of Triple-A game titles.

Used effectively, middleware can help game development companies lower costs by providing solutions that address primary challenges—including lighting. In a recent GamesIndustry.biz interview, Gary Lewis, chief executive officer of Geomerics, noted, "I think every industry is looking at costs," he said. "Even the games industry has to be very aware of what they're spending on games. We're fortunate in the fact that the industry is still very buoyant and still growing at a tremendous rate—but we have to be aware that this probably will not continue at the extent that it has. I'm

sure developers and publishers are looking at the overall costs, and they're looking at areas they can save—and Enlighten would help them."

Productivity is at the heart of the issue. "Enlighten provides a massive boost to productivity, especially when paired with the Intel® multi-core architecture," said Julian Davis, chief technology officer of Geomerics. "Game developers using Enlighten will love the higher fidelity and faster pre-computes. Just switching to the Intel® Compiler gave us a 20 percent performance increase, and using Intel® Threaded Building Blocks our developers were rapidly able to multi-thread the code in a scalable, future-proof way, giving four to eight-fold speed ups on current Intel multi-core hardware."

To explore the possibilities offered by Geomerics Enlighten, visit www.geomerics.com/products.htm. To arrange an evaluation, send e-mail to sales@geomerics.com.





IDV - SpeedTree

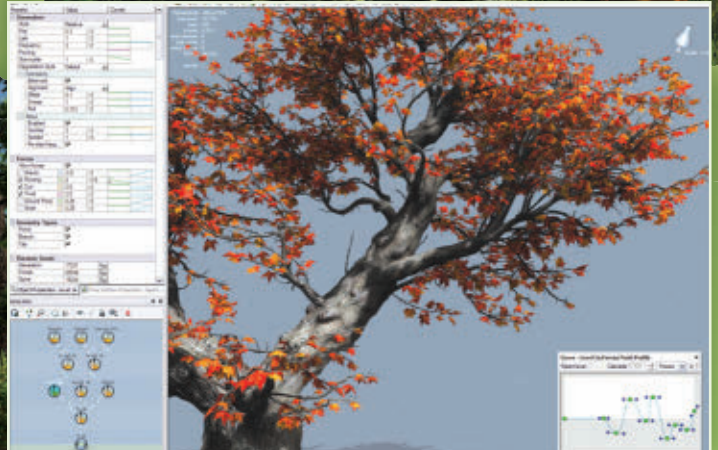
The jungles, medieval forests, city parks, and tropical gardens of the gaming world get a major injection of realism from SpeedTree*.

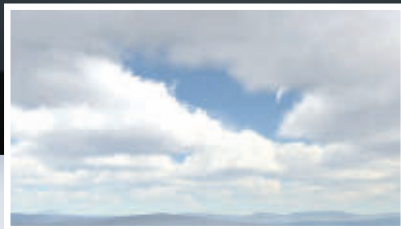
SpeedTree, a product of Interactive Data Visualization (IDV), is middleware that delivers lush, natural real-time trees and plants with seamless level-of-detail transitions; an array of lighting, physics, and wind effects; and an SDK programmed to support any level of engine integration. SpeedTree can be integrated by means of simple mesh export, through integrations with a number of popular game engines, as well as partial or full runtime integration with custom engines.

Optimization work for multi-core platforms, including the Intel® Core™ i7 processor, has dramatically reduced tree computation times for artists using the new tree Modeler, essentially doubling performance when moving from two to four cores. Game developers achieve a performance boost at runtime using SpeedTree when targeting systems featuring the Intel® 4 Series Express Chipsets, such as the GMA X4500, because of optimization and tuning work.

A product with a name like SpeedTree has to be fast. To that end, IDV was an early adopter of Intel® Parallel Studio. "The first time I recompiled SpeedTree using Intel Parallel Studio, performance dramatically improved," said Chris King, CEO of IDV. "I thought I was measuring it wrong. Since then, we've been getting 35 percent speed-ups in the CPU-critical sections of the SpeedTree runtime due to [Intel®] Parallel Composer and [Intel®] Parallel Amplifier. We are completely hooked on the Intel® Compiler, and it has become a permanent part of our development environment!"

For more information about SpeedTree, visit:
www.speedtree.com.





Simul Weather

The Simul Weather*SDK uses the physics of light to create stunningly realistic 3D weather effects.

Game realism benefits tremendously from realistic sky images. Taking maximum advantage of the capabilities of any available hardware platform, this middleware streamlines processor-intensive operations involving lighting and volumetric effects that change over time, such as the movement of clouds.

Lighting and atmospheric effects change significantly throughout the day and—without a middleware solution—game developers can easily expend hundreds of hours trying to capture the appearance of the sky in a convincing way. Simul Weather has captured the sky calculations, including accurate distance-fades, with drop-in source code for sky and cloud renderers. Sample programs using DirectX* and OpenGL* clarify the process.

Parallelism contributes substantially to the real-time responsiveness of 3D weather effects, accelerating the calculations that perform cloud modeling and other weather

phenomena, such as violent thunderstorms. Collaboration with Intel helped unlock the optimal performance of Simul Weather on multi-core processors.

“The Intel® Threading Building Blocks were surprisingly quick and simple to implement, and made the Simul Weather SDK really fly on the Intel® Core™ i7 processor, with close to linear scaling,” said Roderick Kennedy, principal and founder of Simul Software. Simul Weather and Intel’s tools open up great opportunities for game developers to integrate dynamic weather and clouds.”

To view screenshots and videos of spectacular cloud formations, go to www.simul.co.uk/weather/media.



Trinigy Vision

The Vision* Engine from Trinigy gives game development teams around the world the technical and creative freedom to transform their imaginative ideas into immersive gameplay.

Designed to simplify workflow and eliminate the technical barriers that so often hinder game development, this high-performance game engine includes a full-featured, mature toolset and fluid integrations with major third-party middleware solutions. The Vision Engine is available on all major platforms and has been used in over 100 game titles.

The integration of Havok Physics™ into the Vision Engine has captured the attention of the game developers across the globe. As Jeff Yates, vice president of product management at Havok, noted, "... we genuinely believe it will enable Trinigy/Havok customers to experience the most comprehensive physics and game-engine integration available in the commercial market."

"We're really excited to have built-in support for Havok Physics," said Fabian Røken, co-founder of Trinigy. "Integrating Havok Physics into the Vision Engine with Havok's competent and highly experienced engineers will

make it far easier for developers to take advantage of the extensive features in both solutions."

Intel® VTune™ Performance Analyzer and Intel® Thread Profiler have proven invaluable in the threading and optimization work for the Vision Engine, as well as the use of Intel development platforms for testing and quantifying the engine's performance on up to eight cores.

"It is a pleasure to work with Intel," commented Dag Frommhold, managing partner at Trinigy. "Their developer support is highly responsive and has given us valuable feedback on how to optimize the Vision Engine for Intel's latest CPUs and graphics hardware."

To check out the latest developments or to request an evaluation kit, visit: www.trinigy.net.

IMAGE FROM ARCANIA - A GOTHIC TALE ©2009 SPELLBOUND ENTERTAINMENT

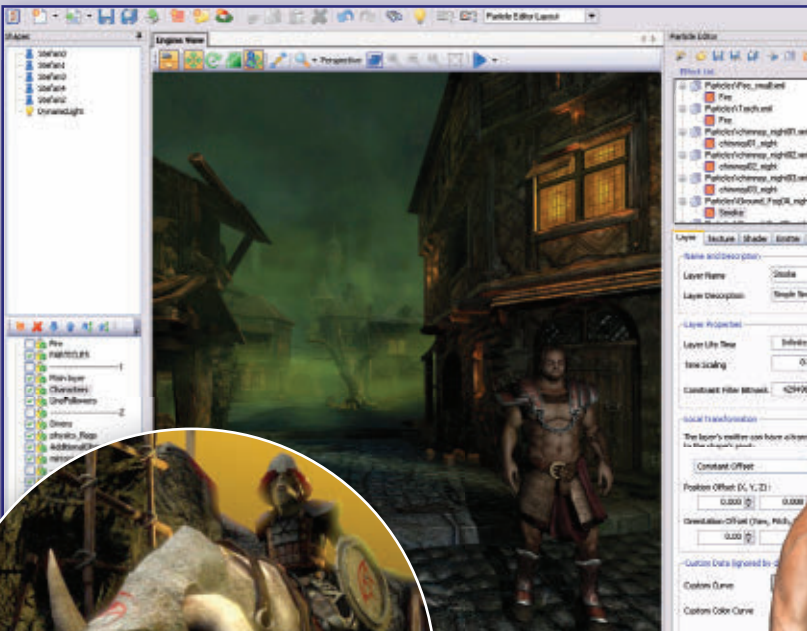
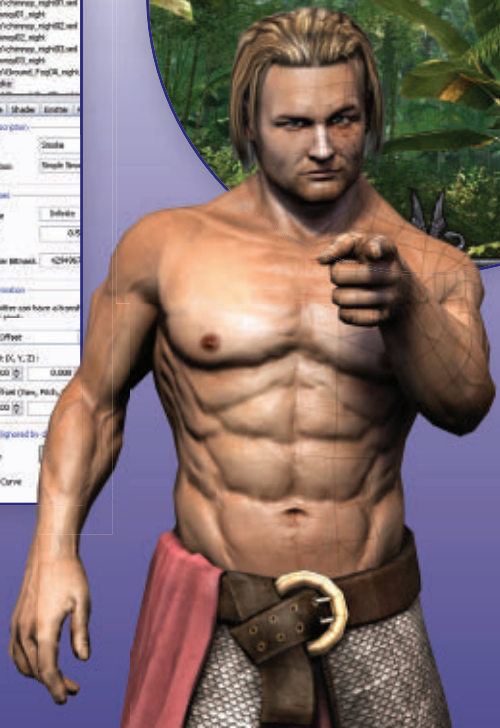


IMAGE FROM DUNGEON HERO ©2009 FIREFLY STUDIOS





PROJECTURF PROJECTURF.COM

REVIEW BY SHEKHAR DHUPELIA

WHEN IT COMES TO PROJECT management software, the king of the hill for some time now has been Microsoft Project. In my experience, many producers, development directors and project managers in the game industry start with this product, or graduate to it from simpler Microsoft Excel tracking sheets (for managing tasks, assets, and so forth), and tend to settle in and adapt it to their needs. However, as I've

detailed in these pages before (see "Tool Box – Microsoft Project 2007," August 2008), Microsoft Project can be incredibly cumbersome, and downright clunky. Further, the tool still focuses on waterfall-style development, doing little to address newer methodologies, such as Agile/SCRUM or Extreme Programming (XP). While Project also has many great benefits, its gaps have opened up an opportunity for several

competitors, both established and newer companies, to introduce their own takes on managing small and large development teams.

Enter Projecturf.com (or more simply just "Projecturf"). Projecturf is an entirely web-based project management tool meant to offer simple and fast project, task, and team tracking for an individual project or across an entire enterprise. The service is completely hosted online to eliminate the need for administration and maintenance, and it is offered at three different price points, to meet the needs of a variety of different organizations. At first glance it's obvious that this is a slick, attractive product, with some of the great collaboration features found in SharePoint and other products. But does it work as a good tool for managing a game's development?

administrator, I found the site to be very straightforward and dare-I-say even fun to use.

That's not to say the service is bug-free. I noticed some bugs, such as when you open up the calendar widget to choose a date, you can't close the widget if you realize you don't want to make a selection yet, and would rather come back to the field later (or never, if it's an optional field). I also was able to layer multiple widgets on top of each other, for example by clicking in one date field and then clicking in the date field right above it.

More troublesome was that the search function seems unusable; it completely missed milestones that contained my search keywords, for example, such as "Concept" in "Concept / Greenlight Review". I expected the search to work like Google, Gmail, or Apple's Spotlight, where I can basically use search as a "crutch", if you will, and just find everything by key word instead of worrying about submenus and folders, but I still found myself having to navigate to things the same as any other software.

A small but cool feature of Projecturf is the built-in customization controls. You can upload a company logo, and specify a color set / theme to the service, so that you and your users experience a site that can look a little more like your existing Intranet or Wiki site. As users navigate the site, it's your company name and logo splashed everywhere, not Projecturf.com. I'd love to see this service expand its offerings to allow a user to make more advanced customizations, such as specifying my own HTML color codes or images for navigational bars or buttons.

USER-INTERFACE

» The UI of Projecturf is very clean and pleasant, and takes some obvious design cues from Apple's OS X and many of the best examples of "Web 3.0" sites. The Projecturf site responds quickly to button presses and events, and has a lot of nice scripting which provides pop-up widgets, form validation on the fly, and many other niceties we've grown used to in web applications, and downright demand in traditional desktop apps. Automated emails from the service come through very quickly and are nicely formatted and straightforward.

Page headers, form fields and labels on the different controls on each page are very clearly detailed, and generally it's very easy to understand what's expected of you on each page and on each form. From the standpoint of a user or an



CONTINUED ON PAGE 40



CONTINUED FROM PAGE 39

FUNCTIONALITY

» Projecturf follows a very straightforward project management paradigm:

- ❏ A project or a game or a feature can be called a PROJECT.
- ❏ The deadlines needed to finish the PROJECT are known as MILESTONES.
- ❏ MILESTONES are achieved by completing TASKS.
- ❏ TASKS are assigned to USERS.

This is easy to learn, and easy to master. There are nicely-labeled buttons and menus to access any of the above, and changes automatically apply across the entire project.

I built a small sample project, created a milestone, and created tasks with deadlines, assigned to users, within minutes. Anyone with basic Internet savvy should be able to pick this up in no time. Updating and changing these details on the fly was also straightforward; you can see a master milestone list or task list under each project, and click on each one to bring up the details and edit whatever is necessary.

A great feature is when you click on "Tools" on the top menu; it brings up a simple Gantt chart that is quite easy to read, and much friendlier than the default view in Microsoft Project. I was able to click on any milestone or task and make changes on the fly, and then go back to this chart. One downside is that you cannot create tasks directly in the chart, as you can in Microsoft Project, nor can you drag and drop tasks, milestones or dates around on the fly. All of this must be done by bringing up each task one-by-one, which can make large, wholesale changes more time-consuming. This also misses out on the opportunity to play "what if?" scenarios, which is a frequent activity in Microsoft Project.

Also in the Tools menu is a "Project Completion Report." This is a beautifully formatted dynamic page that would be great for presenting summary or detailed project information on a projector or in a meeting situation. Bonus points must be given for making this nice

Projecturf PROJECTURF.COM

❏ **STATS**
811 N. Catalina Avenue
Suite 3006
Redondo Beach, CA 90277
www.projecturf.com

❏ **PRICE**
Start-Up—\$29.99/Month
Business—\$59.99/Month
Enterprise—\$159.99/Month

❏ **SYSTEM REQUIREMENTS**
Web connection (Cable/DSL or higher suggested)
Modern browser (IE7-8, Safari, Firefox, Opera)

❏ **PROS**
1 Easy-to-use, attractive and snappy interface.
2 Low-cost bridge product makes a nice step up from simple task lists.
3 Automatic backups and unlimited storage space.

❏ **CONS**
1 Lack of dependencies or auto-milestone calculation.
2 Can't export project data out to CSV or other neutral formats.
3 Geared more towards art and design, not software teams.

report printable directly from this page, into a format that I could never quite seem to nail in Microsoft Project without a lot of customization. Projecturf.com has figured out exactly what the typical producer wants to walk into a meeting with.

One issue I found, once I had some data in the system, is that I could not seem to export any of the data out of the system. They do offer RSS feeds and email notifications for every change within a project, so users know when something's been added or updated, but the data seems otherwise "locked-in." I didn't see a way to pull all of the data out into CSV or other formats, which Microsoft Project and other solutions have offered for some time. This makes it difficult to do any of your own data-mining with custom reports. However, this tool is clearly not meant to replace a high-end project management solution for all industries. This tool seems to be mostly focused on creative professionals. Some examples:

- ❏ A construction manager could not track materials and supplies, as I did not see any way to set the system "resources" instead of "users."
- ❏ There seems to be no way to assign tasks to a sub-set of users; it's all of them, or one of them. This also doesn't fit well with Agile/SCRUM development, and Extreme Programming.
- ❏ The biggest hole seems to be the lack of dependencies. You cannot force a task to start after another task is complete, meaning you can't track real-time slips in your schedule, without manually going in and adjusting tasks to new start dates. Further, milestones don't seem to be dependent on tasks, so you can't get automatic prediction on when a milestone will complete, based on its tasks.

COST / LICENSING

» At the time of this writing, the pricing plans for Projecturf were:

- ❏ **Start-Up: \$29.99 per month—this allows you to manage 15 projects at once.**
- ❏ **Business: \$59.99 per month—this allows you to manage 40 projects at once.**
- ❏ **Enterprise: \$159.99 per month—this is for an unlimited number of projects.**

All options allow an unlimited number of users to be set up, with unlimited storage and a variety of other features included, regardless of which pricing plan you choose.

For developing a single game or cluster of related games, I can't imagine needing anything beyond the "Start-Up" plan. However, if you are more interested in the collaboration features of Projecturf, for example to manage art approvals internally with a centralized art or animation department, or possibly to manage the back and forth flow of assets or code from an outsourcer, then you would likely want to partition the work among many projects. The service is very flexible in that you can choose to upgrade/downgrade or even completely cancel your plan at

any time, with no contracts to worry about, so the choice is always yours.

MILESTONE

» All in all, Projecturf.com isn't the most feature-rich, deep project management tool for complex video game development or major software or hardware projects. However, it is very intuitive and easy-to-use, and certainly the most accessible and pleasant of any tool I've used in this category. This would be a great starting-off point for someone getting their feet wet with moving beyond simple task lists, and would certainly be sufficient for smaller, more nimble game teams.

The site was always snappy and responsive, as fast as any other site I use on a daily basis, and certainly faster than many of my desktop applications. Calling the service attractive or stylish doesn't paint the picture; it's the most pleasant experience I've had with this sort of tool, that's for certain. It is concerning that I was able to find some basic usability bugs in my first few minutes with the software, and it makes me wonder what other bugs I might find on a longer-term basis. Since this is a web-hosted service, hopefully these issues can be fixed easily and "on the fly."

With a few additional features, Projecturf.com could easily mature into a replacement for many off-the-shelf project management packages. In its current form, I would again recommend it to smaller game teams, or to art, animation or design shops (especially outsourcers), who could use this tool to easily track assets and approvals in a light, clean way.

The company tells us that it is preparing to launch Projecturf 2.0 which will include a host of new features and improvements including an enhanced search function, the importing and exporting of project data, assignable items for contact groups, as well as email reminders and nested subtasks.

SHEKHAR DHUPELIA has been in the game and gaming industries since 2000, focusing on game and technology development at companies like Sony and Midway. He is now a producer at WMS Gaming in Chicago.

GDC Canada

Learn. Network. Inspire.

Game Developers Conference® Canada

May 6-7, 2010

Vancouver Convention Centre | Vancouver, BC

Visit www.GDC-Canada.com for more information





FIRE CONTROL

AIM ASSIST IN CONSOLE SHOOTERS

AIM ASSIST IS A STANDARD PLAY CONTROL

element of modern first- and third-person console shooters. In this column I will discuss the concepts behind aim assist and describe one possible implementation.

The term “aim assist” is not universal. Sometimes called “auto-aim” or “reticle magnetism,” I use it to refer to the automatic adjustment of the player’s camera orientation to assist in aiming. It is distinct from a “sticky reticle,” which is the damping of the player’s camera control when the reticle is over a target. The two techniques are often used in tandem.

There are two compelling arguments for including aim assist in your console shooter.

- ❌ **Aiming with a thumb stick is hard.**
- ❌ **Play control logic should do what the player wants.**

AIMING WITH A THUMB STICK IS HARD

» Analog thumb sticks are imprecise measuring devices. Because they must survive a gamer’s wrath and neglect, they are far from the delicate scientific instruments that we wish they were when coding play control.

Thumb sticks do an excellent job when the player is specifying a direction, but they are miserable at communicating subtle magnitude. When a play control mechanism places emphasis on the stick’s exact offset from its center rest position, it is very challenging to get a consistent and responsive feel.

Few controllers report zero magnitude at their rest position. Each controller reports a different magnitude at rest, and the same controller will report different magnitudes each time it returns to rest. Additionally, different controllers report different maximum magnitudes when pressed fully in a direction, and the same controller will report different maximum magnitudes when pressed in different directions.

Despite all this, your game needs to feel the same every play session and on every controller. To account for the variation, we add dead zones. Anything close to the rest position counts as zero magnitude. Anything near maximum counts as the maximum magnitude. These dead zones reduce the thumb stick’s already small range of expression.

Within this small range, the player must be able to express subtle aim corrections as well as large and rapid changes in direction. He needs



FIGURE 1 (top) A console thumb stick’s useful range of expression. About an eighth of an inch.
FIGURE 2 (bottom) The range of expression available to PC shooters. About five inches.

to be able to line up a head shot with the same controls that allow him to do a quick 180 degree turn. All of this is in contrast to the much greater range of motion available to control PC shooters.

The console gamer is clearly going to need some help.

DOING WHAT THE PLAYER WANTS

» A large part of play control logic is determining what the player wants the avatar to do. To create a game that is both responsive and not frustrating to play, we need to do what the player wants, even when what they want isn’t technically what they asked for.

The classic example is the late jump. The difference in response between pressing the jump button while on the ground versus in the air is dramatic. On the ground, the avatar jumps. In the air, nothing happens. So, what do we do when the player presses the button five frames after walking off a ledge? We have the avatar jump, even though it is in the air and that doesn’t necessarily make sense. Jumping is what the player meant when they pressed the jump button. So, as much as is possible, that’s what we do.

Along the same lines, consider a player that has an enemy directly behind his reticle. If that player presses right on the movement-controlling thumb stick, what does he want to happen? He probably doesn’t want to simply move right, which would

cause the reticle to leave the enemy. Rather, he wants to maintain his aim while moving. He wants to circle strafe. By incorporating aim assist into our camera control logic, that is what we give him.

Aim assist is particularly helpful in games where the player is expected to move and aim simultaneously. It is less critical in games where the player is usually stationary while he aims, such as cover-based shooters.

AIM ASSIST

» The fundamental concept behind aim assist is adjusting the player’s aim direction to counteract how that aim would be affected by his movement. With the algorithm at full strength, the screen position of a target is automatically maintained through arbitrary movement of the avatar. The avatar can strafe, jump, move forward or backward, and targeting is maintained. In a sense, it partially decouples the movement and aim controls by coupling the movement and camera controls.

When the player strafes left with a motionless target under the reticle, aim assist rotates the camera clockwise. Likewise, when the player moves right, the camera is rotated counterclockwise. In both cases, with the correct amount of rotation, the target’s screen position under the reticle is maintained.

Despite the fact that aim assist is sometimes referred to as reticle magnetism, the algorithm does not pull the reticle toward targets. In certain situations, the reticle is actually pulled away from the target. Consider a player strafing right across a stationary target that starts to his right. As the enemy’s screen position approaches the reticle, the camera begins to rotate clockwise. Note that this causes the camera to turn away from the target. The speed of rotation is reduced for targets not directly under the reticle, so the rotation does not prevent the target from entering the reticle; it merely slows the approach. While the target is directly under the reticle, the rotation speed is at its maximum. If the player continues right, the target’s screen position leaves the reticle and the rotation tapers off.

DETERMINATION OF AN AIM TARGET

» The starting point of the aim assist algorithm is guessing which target the player is attempting to aim toward. This can be more of an art than a science, and is where aim assist is most likely to go wrong.

The specifics of the choice vary with the details of the game. Not surprisingly, I have had success favoring targets near the reticle over those farther away, and targets at close range over those at long range. The choice should not flicker between targets, but be consistent frame to frame. Game details that can affect the choice include the shape and size of enemy silhouettes and the relative importance of different enemy types. You may want to include things other than enemies in your selection, such as boss weak points and environmental targets (the ubiquitous exploding barrel).

Once determined, the player's aim target is characterized by a set of values that are the inputs of the aim assist algorithm: the position of the target's center, that position on the previous frame, and something I refer to as the target's "centeredness."

Centeredness characterizes how near the target is to the reticle in screen space. It varies between 0 (the target is outside the reach of aim assist) and 1 (the target is near the reticle). A good rule of thumb is that a target with a centeredness of 1 should be close enough to the reticle that it would be hit if the trigger were pulled. Centeredness should vary smoothly with the enemy's screen position.

CAMERA PARAMETERIZATION

» Before discussing the specifics of the algorithm, we should explicitly state the camera's parameterization.

The rendering engine requires that we supply a camera in the form of a homogeneous transformation matrix built from a 3x3 rotation matrix M_C and a position vector p_C . However, that is an awkward parameterization when writing camera control code. I have found a more natural parameterization is spherical coordinates around a camera fulcrum.

The camera fulcrum position is offset from the player's avatar by a distance that usually changes with the avatar's state (standing, crouching, driving, etc.). The variable `camera_theta` is the camera's angle down from the horizon (or the upward angle of the camera's boom), while `camera_phi` specifies the horizontal angle from the world's z-coordinate. The distance from the fulcrum point to the camera is the `boom_length`. In this parameterization, the distinguishing feature of a first-person relative to a third-person shooter is zero `boom_length`.

The camera's rotation and position are calculated from this parameterization as follows.

```
Vector camera_z = Vector(
    cos( camera_phi ) * cos( camera_theta ),
    sin( camera_theta ),
    sin( camera_phi ) * cos( camera_theta ) );
M_C.x = cross_product(
    Vector( 0, 1, 0 ),
    camera_z ).normalize();
M_C.y = cross_product( camera_z, M_C.x );
M_C.z = camera_z;
p_C = fulcrum_pos + boom_length * camera_z;
```

AIM ASSIST ALGORITHM

» The output of the aim assist calculation is a set of deltas for `camera_phi` and `camera_theta` that we apply each frame.

I find it easiest to think in terms of a static camera and moving targets. From that perspective, the effective movement of a stationary target caused by the fulcrum's movement is opposite the fulcrum's movement.

```
Vector effective_target_movement =
    -( fulcrum_pos - previous_fulcrum_pos );
```



FIGURE 3 Aim assist rotates the camera opposite the player's movement.

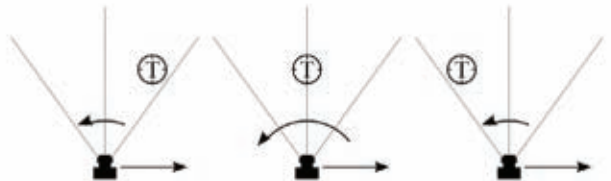


FIGURE 4 The magnitude of rotation varies with the target's vicinity to the reticle.

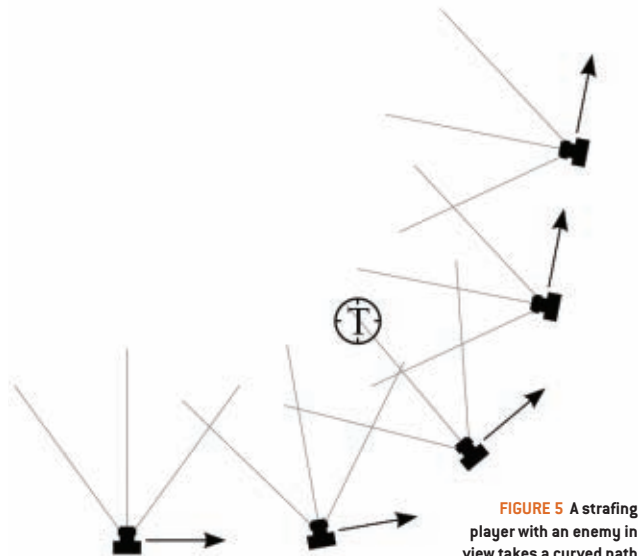


FIGURE 5 A strafing player with an enemy in view takes a curved path due to aim assist.

Note that fulcrum movement originates from both the avatar's movement and changes in the fulcrum's offset from the avatar. For example, the offset may be changing because the avatar recently transitioned from a standing to a crouched state.

We can also incorporate the movement of a non-static enemy. If we add that movement directly, the aim assist will generate camera movement even when the player is not touching the controller. That is not generally desired. Simply zeroing the target's movement when there is no controller input works fine, but you can also incorporate smoother damping algorithms.

```
Vector target_movement =
    target_pos - previous_target_pos;
target_movement = damp_based_on_lack_of_player_input(
    target_movement );
effective_target_movement += target_movement;
```

The next step of the algorithm is to build a reference frame at the target's



position, with the z-component pointing away from the camera fulcrum and with a horizontal x-component. That puts x and y tangent to the surface of a sphere centered at the fulcrum and with a radius equal to the distance between the target and the fulcrum.

```
Vector target_frame_z =
    ( target_pos - fulcrum_pos ).normalize();
Vector target_frame_x = cross_product(
    Vector( 0, 1, 0 ),
    target_frame_z ).normalize();
Vector target_frame_y = cross_product(
    target_frame_z,
    target_frame_x );
```

We express the target's movement in this frame. Movement in the z direction is irrelevant to targeting. It is movement in the x and y directions that we must account for in our aim assist calculation.

```
float lateral_movement = dot_product(
    effective_target_movement,
    target_frame_x );
float upward_movement = dot_product(
    effective_target_movement,
    target_frame_y );
```

Looking first at the lateral movement, we calculate the change in `camera_phi` required to maintain the target's screen position.

```
Vector to_target = target_pos - fulcrum_pos;
float horizontal_distance_to_target =
    sqrt( to_target.x * to_target.x +
    to_target.z * to_target.z );
float delta_camera_phi = -atan2(
    lateral_movement,
    horizontal_distance_to_target );
```

Similarly, we calculate the change in `camera_theta` required to account for the upward movement.

```
float distance_to_target = to_target.length();
float delta_camera_theta = -atan2(
    upward_movement,
    distance_to_target );
```

Notice that the distance in the arctangent is the full distance to the target, while in the calculation for `delta_camera_phi` only the horizontal projection of that distance is used.

If we apply these deltas to `camera_theta` and `camera_phi`, the enemy will maintain its screen position through arbitrary movement on the part of the target and the camera's fulcrum. This is a useful test case. In practice we cap the deltas to a set of maximum angular velocities. These caps become your aim assist strength parameters, and they must be tweaked to your (and your designer's) liking.

```
delta_camera_phi = min(
    max_assist_rot_vel_phi * frame_duration,
    delta_camera_phi );
delta_camera_theta = min(
    max_assist_rot_vel_theta * frame_duration,
    delta_camera_theta );
```

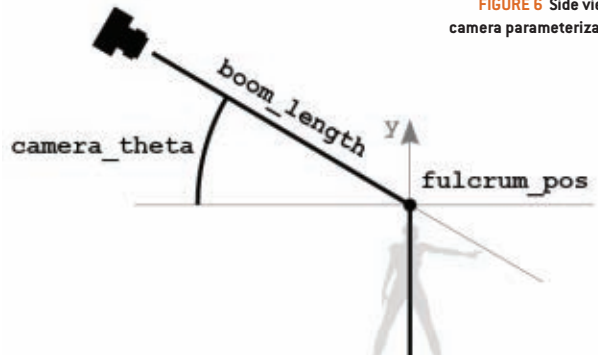


FIGURE 6 Side view of camera parameterization.

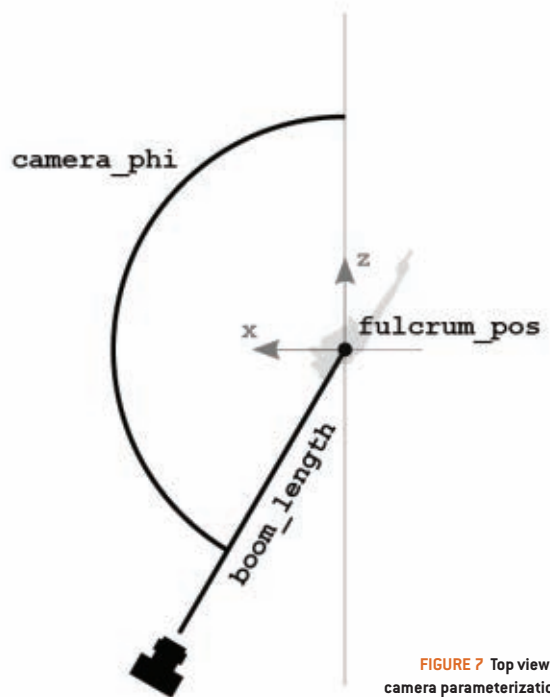


FIGURE 7 Top view of camera parameterization.

To modulate the assistance as the target moves closer and farther from the reticle in screen space, we modulate the deltas by the target's centeredness.

```
delta_camera_phi *= target_centeredness;
delta_camera_theta *= target_centeredness;
```

These are our deltas for the frame. We add these deltas to our camera angles each frame, alongside any deltas generated by the player's controller input.

```
camera_phi += delta_camera_phi;
camera_theta += delta_camera_theta;
```

It is a minor point, but I stated above that the aim assist algorithm, when at full strength, maintains the target's screen position for arbitrary movement on the part of the fulcrum and target. Strictly speaking however, that is only true when the camera's boom length is zero. What

is actually maintained is the angle that the camera would need to be rotated to point directly at the target. In the case of a first-person shooter, that is the same as screen position. For a third-person shooter, the two are slightly different.

Downsides

» There are a handful of situations where aim assist is more annoying than helpful.

The obvious case is when you guess the player's desired target incorrectly. Consider a player strafing to line up a shot at a distant enemy. If a closer enemy moves on screen, it is picked up by the aim assist in preference to the farther target. This is particularly awkward if the new enemy appears opposite the side the player is strafing towards. The reticle is pulled away from the intended target, toward the new target, in a way that is both obvious and annoying.

Another irritation caused by aim assist occurs when the player pulls the reticle off of a target while strafing in the same direction he is rotating the reticle. The reticle receives a burst of speed when it leaves the target, almost as if it were being kicked away. This burst of speed is a result of the aim assist no longer fighting the player's camera rotation. The effect can disorient the player as he adjusts to the higher speed. While a smooth fall off of centeredness should mitigate the issue, I have never been successful at eliminating it entirely.

Situations in which the player can move at high speed are problematic. Because a high rate of camera rotation is required to account for the player's speed, you must make a choice between having effectively no aim assist and allowing aim assist to produce disorientingly large rotation rates. One way to mitigate this issue is to design vehicle encounters so that enemies stay a good distance away from the player. If the ratio of player speed over enemy distance is kept consistent between encounters, a single aim assist strength will work throughout the game.

Another characteristic of vehicles which can cause issues is their often high camera fulcrum. When the fulcrum is significantly higher than the height of targets, the player must adjust his aim vertically based on an enemy's distance. As the player approaches an enemy, aim assist pulls the reticle down into the ground. When the player engages a new enemy, he must drag his reticle back up to horizontal. This quickly becomes tiresome. Similar to high player speed, the issue is mitigated if vehicle encounters are designed such that enemies keep their distance.

Aim Assist as an Option

» Most players do not notice aim assist. They enjoy games more because of it but are not conscious of its effects. A small minority of expert gamers are aware of aim assist, and a subset of them deplore it. This leads to the question, should aim assist be optional? If so, do you take steps to ensure your game is fun without it; perhaps increasing the time enemies are stationary, or reducing the amount of dodging required of the player? If not, you have given the player the option to make your game less fun. Those who choose that option will blame you, not themselves, if they find your game frustrating or overly difficult. If aim assist is made optional, it should default to on. Options should default to the setting appropriate for a casual player.

Despite aim assist's pitfalls and detractors, it is a de facto standard in console shooters. Incorporating the technique into your shooter's play control is likely to improve the players' experience. 🎮

DANIEL NELSON has been a play-control, animation, and physics programmer at Neversoft Entertainment for seven years and is credited on the *TONY HAWK, GUN,* and *GUITAR HERO* series. He has a Ph.D. in high-energy physics from The Ohio State University.

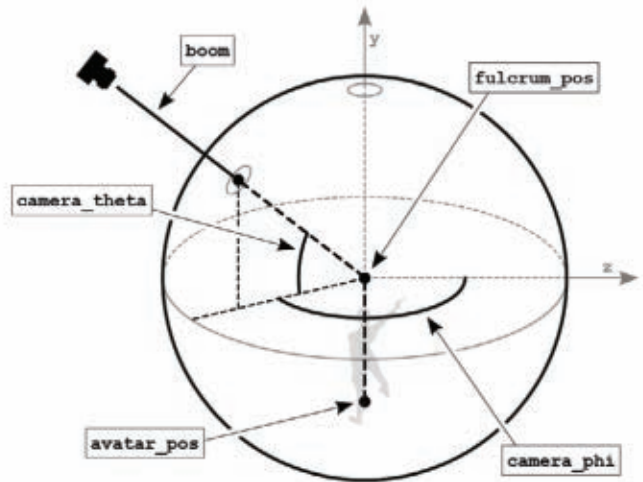


FIGURE 8 Isometric view of camera parameterization.

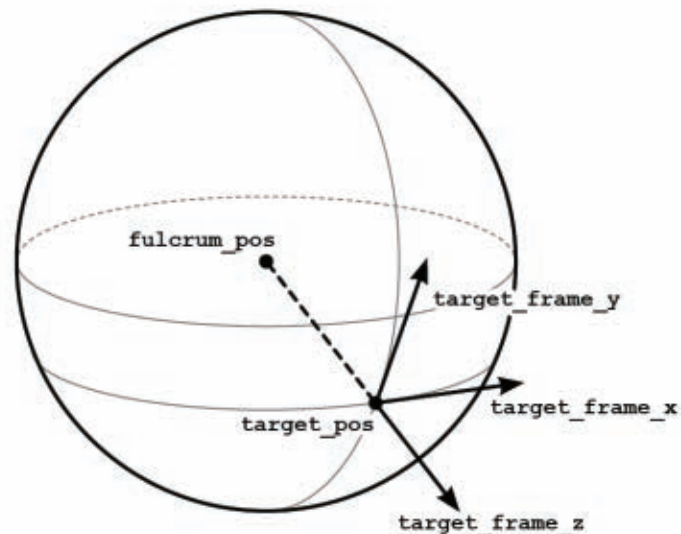


FIGURE 9 We build a reference frame at the target's position.

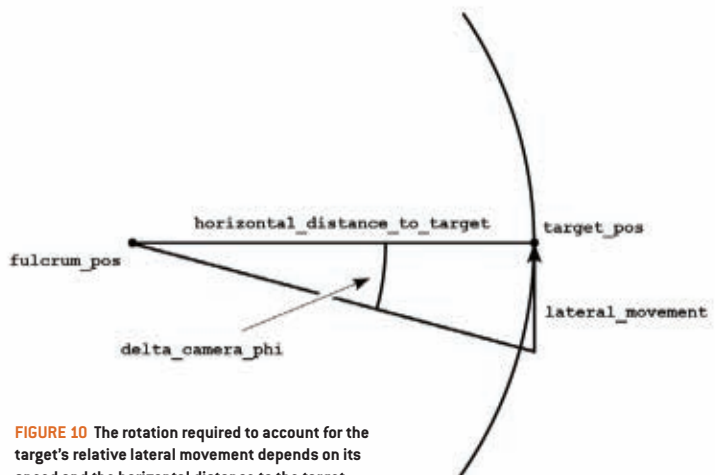


FIGURE 10 The rotation required to account for the target's relative lateral movement depends on its speed and the horizontal distance to the target.



UNKNOWN UNKNOWNNS

THE ART AND SCIENCE OF THE SCREWUP

“..there are known knowns; there are things we know we know. We also know there are known unknowns; that is to say we know there are some things we do not know. But there are also unknown unknowns—the ones we don’t know we don’t know.”

—Donald Rumsfeld

IT’S THOSE UNKNOWN UNKNOWNNS THAT REALLY get you. The other day my wife came bursting through the door, flustered. “Where’s the car? Did you move the car? I think somebody stole the car!”

“What, it wasn’t by the bus stop where you left it this morning?” I replied.

“Oh,” she answered sheepishly. “Oops. That’s right. I drove to the bus this morning...”

You’ve probably got a similar story. I’ve made the exact same mental leap—more than once. So has every other person I asked about it.

This little incident illustrates something basic about human nature: we’re oblivious to our mistakes. Psychologists have examined this phenomenon in hundreds of different ways and always come to the same conclusion: people have a cognitive blind spot for slipups, memory lapses, and trivial errors.

This isn’t just vanity or pride. Even guilt-ridden types who shoulder the blame for everything big are as likely as the rest of us to overlook their own unconscious omissions. Self-flagellators and arrogant jerks are equally unlikely to realize

they’ve forgotten where they parked the car. Both may accept intellectually that people forget where they’ve parked far more often than cars get stolen... but both are equally likely to freak out when they’ve mentally filed the car in one spot and it’s physically parked somewhere else. When an unknown is unknown, it’s extremely hard to account for. By definition, you don’t know what you don’t know.

Of course, the illusion of infallibility doesn’t just cover cars. Even brilliant, charismatic, and accomplished people—the sort, say, who work as game artists—fall victim to it all the time. Just ask any tech artist or tools programmer: the vast, vast majority of “bugs” that artists experience are really just innocuous little mistakes. There’s that file that just won’t export correctly: “It still looks exactly like it did this morning! ... Oh, wait, looks like I forgot to check it out. Sorry.” Or the shader that’s completely busted: “Damn it, my normal maps aren’t showing up ... well, would you look at that! I copied the TGA over from the temp version and forgot to change the file name. My bad!” Or

the ragdoll that flies apart into a million pieces: “Sheesh, looks like I left the mass number set to the default, it only weighs 1 gram.”

TO ERR IS HUMAN

» It’s easy to dismiss these kinds of problems as trivial, because they generally involve such simple and innocent mistakes. It’s also easy to think they come from sloppiness or laziness. But despite what that engineer standing over your shoulder may be mumbling under his breath, these little glitches aren’t a sign of incompetence or stupidity. They’re basically inevitable in a complex world like ours.

Think about it this way: a typical OCR reader is about 98 percent accurate. That sounds impressive. Wouldn’t you be proud to be 98 percent accurate? But have you read much OCR’ed text? When you do lots of stuff, 98 percent isn’t really that great. In this paragraph, for example, there’s an error in 2 of every 100 characters. Turns out 98 percent produces a potty noticeable numbers of misbakes, doesn’t it? And honestly, how many of us are really 98 percent organized or 98 percent focused all the time?

Thus we generate a constant dribble of little mistakes. Each one is trivial, often taking only a few seconds to fix. Unfortunately they add up to a huge—but invisible—drag on our days. If anything else sucked up time and energy like our own little mistakes, we’d be furious. We grumble constantly about our tools—but Max



GTA IV (left) and WORLD OF WARCRAFT (right) glitches.

and Maya crashes are, in fairness, much less common than all the little accidents we inflict on ourselves with typos, source control blunders, and incorrect settings. But we remember events that victimize us unfairly far more clearly than the little bitty blunders we make all the time. The time last month when Max erased all your keyframes, or the day Maya wiped your preferences in the midst of crunch will loom a lot larger in your memory than the twelve times this week you've tried to save files without checking them out.

To make things worse, our innocent little mistakes suck up time and energy out of proportion to their seriousness. Not only are little mistakes easy to make; by definition, they're almost invisible. Thus, they're always the last suspects when it's debugging time. Say you're trying something new and tricky with a shader. You'll be vigilant for evidence that your fancy new technique is working (or barfing). If that shader fails because of something silly, like typo or a missing texture, it's almost inevitable that you (often, with a bunch of other people)

will be poring over the high-tech innards of the system for a quite while anyone notices that you swapped two letters or forgot to copy a file.

OOPS I DID IT AGAIN!

» If it's any comfort, the infallibility effect has been studied for years and it's a universal problem. It's not caused by working in games, overconsumption of pizza at 11 p.m., MMO addiction, nor even sheer stupidity. It's pretty much a law of human nature, which means it's hard to get around. The question is: What are you going to do about it?

The obvious answer would seem to be "stop screwing up." Unfortunately that's a prayer, not a plan. The essence of dumb mistakes is that nobody makes them on purpose. You can plan ahead to minimize risks in things you intend to do—"Last time we tried to combine swappable face textures and facial animations it really worked out badly—let's budget three weeks for proof of concept before we decide to put that on the feature list." But you can't plan around brain farts: "You know, I was planning to mis-type

this file path and then spend an hour wondering where my stuff went—but maybe I'll pass ..."

You can't will mistakes out of existence. Instead, look at folks who really, really have to worry about screwing up. Airline pilots, surgeons, and NASA engineers don't get a second chance if something trivial gets overlooked (Remember that multi-million-dollar space probe that pancaked because half of the guidance code was written in metric and the other half was done in English units?). Disciplines with low risk tolerance have developed a bunch of tools and techniques for making imperfect, absentminded human beings more reliable.

FAILURE TO LAUNCH

» The classic way to avoid dumb mistakes is to double-check your work. Pilots, scuba divers, and surgeons are all trained to use formal checklists so they can't miss a vital step. This sounds like overkill for us: "Files checked out to me? Check! Export flags set correctly in my Max file? Check! Stand by for export. 5, 4, 3, 2, 1—We have Control-Shift E!" Nevertheless, the basic idea is sound—if

MARCH 9-13, 2010

GDC 10

Game Developers Conference®
Moscone Center, San Francisco, CA
www.gdconf.com

**THINK SERVICES
GAME GROUP
EVENT CALENDAR 2010**

<p>MAY 6-7, 2010</p> <p>GDC Canada</p> <p>Game Developers Conference® Canada Vancouver Convention Centre Vancouver, BC www.gdc-canada.com</p>	<p>AUGUST 16-18, 2010</p> <p>GDC Europe</p> <p>Game Developers Conference® Europe Cologne Congress Center East Cologne, Germany www.gdceurope.com</p>	<p>OCTOBER 5-8, 2010</p> <p>GDC Austin</p> <p>Game Developers Conference® Austin Austin Convention Center Austin, TX www.gdcaustin.com</p>	<p>RETURNS IN 2010</p> <p>GDC China</p> <p>Game Developers Conference® China Shanghai International Convention Center Shanghai, China www.gdcchina.com</p>
---	--	---	--



A collision failure in FORZA MOTORSPORT 2.

you can identify a definite set of things that must happen for things to work right, you really ought be able to walk through that list item by item for clarity and consistency.

Checklists don't feel very artistic, but they're a bit easier to tolerate if you think of them as insurance, rather than a rigid procedure to follow every time. Figuring out what all the required steps are, how they relate to each other, and why they're needed will teach you a lot about how things work in your pipeline. Having that list written down somewhere will let you walk through it line by line next time a bug shows up. Not only will it help you get things done right, it'll make sure you don't call in the tech-support cavalry till you can prove the problem is really not your fault. You'll have fewer problems and you'll get better service, to boot.

Of course, paper checklists are passé. Critical, complex operations like creating a new asset or exporting to the game should come with checklists of important steps built right in. An "export wizard" can keep you from doing boneheaded stuff like exporting out-of-date files or putting things into folders your game can't read, and will cut down on a lot of trivial errors. It's a huge improvement on the typical dialog box. Not only does the built-in checklist help catch mistakes, it's also a constant, low-pressure way of reinforcing good habits. Checklists and wizards are popular in commercial software for exactly this reason — they make complex tasks less error-prone and more comprehensible.

BLOOPER REEL

» Maybe—Okay, probably!—you can't wrangle a full-fledged checklist out of your tools team for every important tool. Even non-technical artists can still cut way down on preventable screwups

with simple tricks. Simple things like a batch file that gets the latest files for you with one click can eliminate a lot of possible glitches. If your studio relies on naming conventions, you can keep correctly-named template files and folders around and simply copy them to create new assets, instead of rebuilding the whole structure from scratch every time you create a new asset.

The details of any solution, naturally, are very tied to the nuances of your pipeline; but the basic principles are really simple:

✂ Cut down on the number of manual steps.

More steps means more opportunities to forget something, mis-type something, or just plain screw up. If you're doing stuff like typing the same name in five different places you are asking for trouble (and getting nothing valuable or artsy in return).

✂ Fail loudly. It's much better if your workflow tells you "hey, you're trying to export a file without checking it out! Stop!" rather than allowing you to think you've made a change. You want your mistakes rubbed in your face, not swept under the rug (or, like in most companies, buried in among 700 lines of meaningless error spew in your importer's log file).

✂ Make the feedback useful. This is a balancing act—a tool that fails silently without letting you know something went wrong is just as unhelpful as a tool that buries you under useless information. If you're lobbying for support or building your own tools, keep the feedback simple and make sure to tell the users what should be done ("this file is in DXT5! If you really want to use it as an re-color map, convert it to DXT1").

MISTAKEN IDENTITY

» The moral of this story is pretty simple. <Expletive> happens. It's going to happen and you can't wish it away or pretend that just yelling at people will turn them into precision machine tools. We do a complex, many-layered job, and we're going to screw up sometimes.

However, this doesn't mean we can't make things better. The key thing is to be aware that mistakes are inevitable and design your workflow to be more glitch-tolerant. Don't rely entirely on good intentions. Even if you don't use a checklist every time, have one handy when the debugging phase comes around. Simplify and streamline your workflow to cut down on opportunities for screwups. Design tools with good feedback and wizard-style interfaces to guide users politely down the right path. Make sure that these errors which do happen announce themselves loudly enough to be caught right away.

If you can do all that, you'll still have to deal with mistakes. Our job is too complex, and too full of creative uncertainty for perfection. You can, however, trade up to a more interesting and classier set of mistakes, which is progress of a sort. Plus, mastering the steps you use to cut down on glitches will also free you up to get back to your real job: making art instead of pushing bottoms.

Wait, did I spell that right? 🗣

STEVE THEODORE has been pushing pixels for more than a dozen years. His credits include MECH COMMANDER, HALF-LIFE, TEAM FORTRESS, COUNTER-STRIKE, and HALO 3. He's been a modeler, animator, and technical artist, as well as a frequent speaker at industry conferences. He's currently a consultant helping game studios perfect their art tools and pipelines.



CHALLENGING DESIGN

MANAGING DIFFICULTY AGAINST FRUSTRATION

THE SURGERY GAME TRAUMA

CENTER was one of the earliest examples of how the Nintendo DS could change our industry. By turning the stylus into a scalpel, the designers let players immerse themselves into the role of a doctor as never before. Unfortunately, the game simulated the pressures of actual surgery too well, by presenting staggeringly difficult, time-pressured levels.

Failure blocked the player's progress, which proved to be a fatal flaw for the game because there were no difficulty levels at all—no way for the player to decide what level of challenge was appropriate. Considering the wide demographic of gamers today, from young children to seniors, this decision restricted the game to a smaller slice of the DS's audience.

Challenge has always been a core component of game design. However, after video games left the arcades—in which quick difficulty ramps were a necessity of doing business—most designers realized that their games could appeal to more people if they tailored the challenge to meet the needs of the individual user.

DYNAMIC DIFFICULTY

» CALL OF DUTY 4, for example, measures the player's performance during the training level to suggest an appropriate difficulty level. Other games—such as LEFT 4 DEAD—have developed dynamic difficulty algorithms which adjust enemy spawns

and health drops to the player's current situation and demonstrated skill.

While this is appealing, dynamic difficulty can be a tricky proposition as well. Similar to AI cheating, if the player can see the invisible hand controlling the challenge, the spell is broken. Players need to perceive that they are improving against a fixed measuring stick. The RPG OBLIVION turned off many people by scaling the weapons and skills of enemies directly in relation to the player character's level.

Once this mechanic became obvious, many absurd strategies emerged, such as never leveling up to ensure that enemies always stayed weak. More significantly, this dynamic ruined one of the core features of an RPG—power progression. After developing advanced characters, players enjoy easily brushing aside monsters which earlier in the game could have destroyed them.

ELECTIVE DIFFICULTY

» Indeed, the core mechanic of RPGs—that the player character grows slowly in power after each successful battle—can be seen as a way to adjust the game's difficulty themselves. Gamers who feel comfortable with the combat system can push ahead through levels at the edge of their abilities while players who prefer a more comfortable experience can grind their way to overpowered characters before proceeding. Most

importantly, this system puts the player in control, not the designer.

Although selecting a difficulty level at start was a simple, early innovation, only recently have games allowed players to switch between them during normal play. On every third death in NINJA GAIDEN BLACK, players could elect to drop to "Ninja Dog" mode, which weakened enemies but also forced Ryu to wear pink ribbons as punishment. This mechanic—minus the mockery—was quickly adopted by other games, such as GOD OF WAR. Returning to LEFT 4 DEAD, due to its dynamic AI, players can choose to drop the difficulty down (or raise it!) at any time during a level.

Indeed, elective difficulty itself can be a core gameplay mechanic. The browser-based DESKTOP TOWER DEFENSE has no difficulty levels at all, but does allow the player to speed

beginning, as players must learn how to master the speed-up mechanic to start improving their scores.

ORTHOGONAL CHALLENGES

» Similarly, THIEF determines the difficulty mode not at the beginning of a level but by how the player challenges herself during the level. The requirements of Easy may only be stealing a certain number of jewels and artifacts while Hard also necessitates finishing the level without killing a single guard.

These different modes suggest orthogonal challenges within the same game, a smart way to extend a game's life for the hardcore. Other official examples include the One City Challenge and Always War options in CIVILIZATION 4 and the Hardcore mode (with permanent death) in DIABLO 2. Indeed, Xbox Live

se. For example, STARCRAFT had both a difficulty setting and a speed setting, so a player could try a more difficult AI but at a slower speed if he did not enjoy time pressure. One sadly forgotten setting is the complexity option that appeared in earlier games, such as M.U.L.E. and LORDS OF CONQUEST. This option provided a simpler version of the game—with fewer types of resources, for example—but still with a fully-capable AI that could provide a challenge for new players.

CHALLENGE AND PUNISHMENT

» However, some games choose to punish players on top of giving them a fair challenge. Games without generous save systems, for instance, are vulnerable to being ruined by challenging sub-sections, which might require multiple attempts to pass. If a player needs to

Although selecting a difficulty level at start was a simple, early innovation, only recently have games allowed players to switch between them during normal play.

up the game (and thereby increase the challenge) by triggering attack waves prematurely. The final score is calculated not just from how many enemies were destroyed, but also from how quickly the game finished. Therefore, beating DTD on the default speed is just the

Achievements provide a fantastic infrastructure for adding new challenges via unorthodox goals to games that might otherwise no longer interest core gamers.

Furthermore, other settings can adjust the challenge of a game without changing the difficulty, per

repeat a lengthy but easy section (or, more shamefully, a non-skippable cut-scene) before getting to the difficult bit, the game is punishing the player instead of challenging him.

One of the most elegant solutions to this problem was the time control mechanic



Valve's *LEFT 4 DEAD 2* adjusts difficulty on the fly according to player performance.

in *PRINCE OF PERSIA: SAND OF TIME*, in which the player is able to rewind past mistakes a limited number of times to try again. This system reduced the overhead of repeating a difficult jump to a relative minimum while still retaining tension because of the finite number of rewinds.

Another example of reducing punishment can be seen in the history of MMOs. *WORLD OF WARCRAFT* famously reduced the penalty for death found in earlier MMOs, such as *EVERQUEST* and *ULTIMA ONLINE*. By removing corpse runs and experience loss, *WoW* enabled people to play the game the way they wanted to play it. Instead of only attacking easy monsters which would never cause the loss of experience or loot, players could attempt a difficult battle knowing that, in the worst case, they would be warped back to a safe

location with some minor equipment damage.

Thus, games with severe penalties for failure can actually warp the core gameplay by strongly encouraging players to always choose the safe route. *DEFENSE OF THE ANCIENT*, the popular mod for *WARCRAFT 3*, rewards the opposite team with gold every time a player is killed, which makes bumbling new players extremely unpopular with their teammates. This simple dynamic makes the *DOTA* community notoriously nasty and unpleasant, even by the lowered standards of the Internet.

AFTER-PUNISHMENT

» The strategy/puzzle hybrid *PUZZLE QUEST* took *WoW*'s forgiving nature to the logical extreme by removing all forms of punishment from the game entirely. Players

are even rewarded for losing battles, albeit much less than they would be for winning them. In fact, this mechanic has an interesting side benefit; *PUZZLE QUEST* has no need for a visible save system. Because players are never penalized in any way, the game can comfortably auto-save after every battle or action, knowing that a player will never feel the need to revert to an earlier save.

Such a forgiving system is not for every game. *BIO SHOCK* used a similar mechanic by respawning dead players for free in *Vita-Chambers* placed throughout the game. Furthermore, enemies health rates were not reset on a player respawn, which meant that the player could chip away at any enemy with any weapon, including the wrench, if she was willing to die and be reborn enough times. This feature felt like an

exploit to enough players that *Irrational* eventually patched in an option to disable *Vita-Chambers*.

However, the problem may have been with the expectations of *BIO SHOCK*'s intended audience instead of any fundamental flaw with the respawn mechanic. *LEGO STAR WARS* uses an identical mechanic, which is perfect for the target audience of a dad and a son playing together in a forgiving environment. For *BIO SHOCK*, core gamers expected the game to force them to use advanced strategies to progress instead of an easy out.

Perhaps the best solution is to always allow players to progress but to rate their performance against some constant metric. *ELITE BEAT AGENTS* hands out letter grades of S, A, B, C, and D for each song performance based on the player's timing.

The game continues as long as a player achieves a C, but few players will not want to go back to try and improve. If *TRAUMA CENTER* had only adopted such a simple system, the game may have been much more popular.

Designers should take care not to head down the same dead-end road; knowing that alternate models exist, rather than simply retreading the old "choose difficulty before you play" path, can help introduce and popularize more core experiences to new audiences. 🎮

SOREN JOHNSON is a designer/programmer at EA, working on browser-based strategy games at www.strategystation.com. He was the lead designer of *CIVILIZATION IV* and the co-designer of *CIVILIZATION III*. Read more of his thoughts on game design at www.designer-notes.com.



RALLYING RIGHTS

HOW GAMES ARE (FINALLY) ATTRACTING THE ASCAP'S ATTENTION

VIDEO GAMES ARE THE PRECOCIOUS

toddler of the media world. Young, unfazed by the rules our older siblings seem so confined by, and more than happy to chaotically wander off into new directions whenever we feel like it. At the birth of the game industry's Silver Age—the dawn of the era of PlayStation 2 and Xbox—the established music industry never imagined that games like GUITAR HERO and ROCK BAND would become an interactive means of new music distribution. Nor did the home video market foresee the shift to an environment where movies are delivered digitally via PlayStation Network and Xbox Live. Time and again, the video game industry has changed the way established media is experienced, regarded, and defined. In the wake of our trailblazing, traditional media and the mechanisms it long ago established are scrambling to keep up and adapt to the new world we're creating.

Part of the old media infrastructure chasing gaming's innovations is The American Society for Composers, Authors, and Publishers, or ASCAP as it is more commonly known. After decades of disinterest in the music of interactive entertainment, ASCAP is now actively making a play for the inclusion of game composers into its ranks.

ASCAP GIVETH

» At its core, ASCAP is a performing rights organization that collects money in the form of royalties for the "non-dramatic live performance" of copyrighted music registered with it by its members. In practice, ASCAP defines "live performance" rather broadly and includes everything from television to radio airplay to music played within malls, restaurants, airplanes, and telephone on-hold applications.

When registered music is "performed," ASCAP collects and

distributes writer and publisher shares of the royalties to its members. If music isn't registered with ASCAP, ASCAP Licensees—such as TV networks and radio stations—still pay royalties for its performance. ASCAP then divides any unclaimed royalties up between its registered members.

This year, ASCAP woke up to the spreading influence of games and the notion that game scores no longer sound like those of 30-year-old arcade cabinets. In May, ASCAP reached out to game composers and sound designers. Their intent was to educate music creators about rights and royalties that have long been regarded as either off-limits or too difficult to obtain within the game industry.

With almost all game audio contracts falling into the work-for-hire vein, content creators are used to signing over all rights and claims for a full buy-out payment. ASCAP's point is that film and TV composers are also frequently forced to sign work-for-hire contracts, but that they negotiate for the ability to collect performance royalties. At the heart of the argument is the notion that, while a corporation can claim the publisher share of royalties, corporations cannot claim the writer share. As such, money is being left on the table unnecessarily that could be claimed by the content creator without being at the expense of game developers or publishers. Secondarily, were game development companies and publishers to register themselves with performing rights organizations as the music publisher for a given title, they would open themselves up to a new stream of revenue in the form of the publisher share of royalties.

In order to be able to claim the writer share of royalties, composers would need to first negotiate into their contracts with the game's developer or publisher the ability to register their material with a performing rights



Games such as ThatGameCompany's FLOWER, which is only available via digital distribution, could potentially see itself become a stream of revenue for publisher Sony if ASCAP gets in with game composers. Or would these indie games simply have to pay out to ASCAP in order to exist? Many questions remain.

organization. ASCAP has suggested the following sample legal text as a basis for these negotiations:

"Composer shall be entitled to collect the 'writers share' of public performance royalties [as that term is commonly used in the music industry] directly from a public performance society that makes a separate distribution of said royalties to composers and publishers."

ASCAP TAKETH AWAY

» While ASCAP is now actively trying to welcome game composers into its ranks, it is also trying to expand the definition of "live performance" so as to include a slew of new media applications and thereby widen the scope of licensee companies who pay royalties to their members. Among those new media applications targeted, ASCAP is claiming that digital distribution of games via Xbox Live and PlayStation Network—since it is done via the Internet—now constitutes public performance of the accompanying music scores for downloaded games.

Whether ASCAP will be able to convince publishers and first-party companies like Sony and Microsoft

that digital distribution represents a new, legitimate stream of royalties for its members remains to be seen. In October, ASCAP lost its previous legal claim that telecoms owed royalties for cellular phone ringtones being played in public spaces when a subscriber receives a call.

So far the other major domestic performing rights organization, Broadcast Music, Inc. (BMI), has yet to follow suit and aggressively reach out to game professionals with the carrot of royalties. But rights organizations network with each other across the globe, and successful ventures into new technologies will eventually become standardized practice throughout the traditional mechanisms of the music industry. What is abundantly clear is that game innovations continue to ripple throughout the media world. As they do, more and more opportunities are presenting themselves to the audio professionals who are helping to drive those innovations. [f](#)

JESSE HARLIN has been composing music for games since 1999. He is currently the staff composer for LucasArts.

ACTIVISION®

GREAT GAMES

START WITH

GREAT PEOPLE



ARE YOU READY TO EXPLORE THE MOST SOUGHT AFTER JOBS IN ENTERTAINMENT?



Visit our site: activision.com

ACCESS IT IN REAL TIME

gamedeveloper



DIGITAL EDITION



Game Developer magazine's 6 month and 1 year Digital Edition subscriptions give you new issues delivered promptly, full searchable access to more than 50 back issues, plus downloadable PDF versions, easy access from any web browser, and more!

Subscribe today!

[HTTP://WWW.GDMAG.COM/DIGITAL](http://www.gdmag.com/digital)

THINK
SERVICES
A DIVISION OF UNITED BUSINESS MEDIA, LLC

MAKE MORE ENEMIES

GAME DESIGN AT VANCOUVER FILM SCHOOL shows students how to make more enemies, better heroes, cooler levels, and tighter industry connections.

In one intense year at VFS, you'll learn every aspect of game design by creating playable games of your own.

VFS grads get hired by top companies like BioWare, Radical, Relic, and Ubisoft, and the LA Times named VFS a top 10 school "most favored by video game industry recruiters."

VFS

vfs.com/enemies

VFS student work by Moo Won Kim



THINK SERVICES
GAME GROUP

network of sites

www.gdmag.com
gamedeveloper



GAMASUTRA

The Art & Business of Making Games
www.gamasutra.com

GAME CAREERGUIDE
www.gamecareerguide.com

WORLDS IN MOTION.BIZ
www.worldsinmotion.biz

GAMERBYTES
www.gamerbytes.com

indiegames.com
www.indiegames.com

GAME SET WATCH
www.gamesetwatch.com

GAMES ON DECK
www.gamesondeck.com

SERIOUS GAMES SOURCE
www.seriousgamesource.com

FG FINGER GAMING
www.fingergaming.com

CENTER FOR DIGITAL IMAGING ARTS AT BOSTON UNIVERSITY



capture
imagination

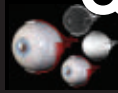
3D ANIMATION
+ INTERACTIVE MEDIA

GAME ART + CHARACTER ANIMATION

..... certificate programs

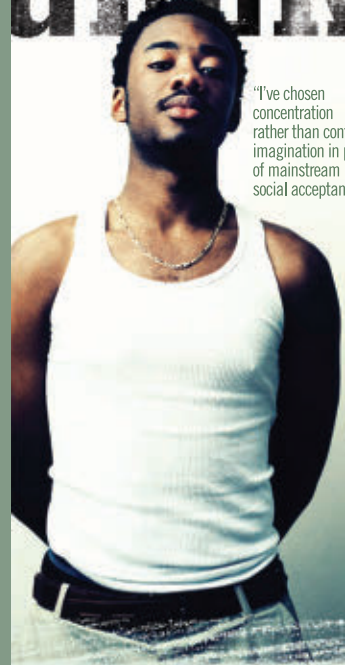
TWO CAMPUS LOCATIONS
WALTHAM, MA & WASHINGTON, DC

Financial assistance and career services available.
Now accepting applications. **Apply today!**



CDIABU.COM

I'M A
GEEK.



"I've chosen concentration rather than conformity, imagination in place of mainstream social acceptance."



Come get your geek on.

UAT is a community of geeks—passionate seekers of knowledge and truth in an ever-expanding universe of technology.

LEARN:

- | | |
|---------------------------------|---------------------------------|
| ADVANCING COMPUTER SCIENCE | NETWORK ENGINEERING |
| ARTIFICIAL LIFE PROGRAMMING | NETWORK SECURITY |
| DIGITAL MEDIA | OPEN SOURCE TECHNOLOGIES |
| DIGITAL VIDEO | ROBOTICS & EMBEDDED SYSTEMS |
| ENTERPRISE SOFTWARE DEVELOPMENT | SERIOUS GAME & SIMULATION |
| GAME ART & ANIMATION | TECHNOLOGY FORENSICS |
| GAME DESIGN | VIRTUAL MODELING & DESIGN |
| GAME PROGRAMMING | WEB & SOCIAL MEDIA TECHNOLOGIES |

www.uat.edu > 877.UAT.GEEK

ADVERTISER INDEX

COMPANY NAME	PAGE	COMPANY NAME	PAGE
Autodesk.....	C3	twofour54.....	C2
Center for Digital Imaging Arts	55	Michigan Film Office	6
Epic Games	15,23	RAD Game Tools	C4
Georgia.org.....	3	University of Advancing Technology.....	55
Havok.....	11	Vancouver Film School	54
Intel	32-38		

Game Developer (ISSN 1073-922X) is published monthly by United Business 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 United Business 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*, 4601 W. 6th St. Suite B, Lawrence, KS 66049. Call toll-free in the U.S./Canada (800) 444-4881 or fax (785) 838-7566. 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/United Business Media LLC, unless otherwise indicated. Don't steal any of it.



PROJECT STATUS UPDATE!

TEAM BOVARY FIRES ON ALL CYLINDERS

ILLUSTRATION BY JUAN RAMIREZ

HEY ALL—IT'S BEEN A WHILE

since we last did one of these, and I just wanted to give you all the bird's-eye view of where things are on the project as a whole and highlight some of the exciting things we have going on. I think it's fair to say our game adaptation of Gustave Flaubert's *Madame Bovary* is really going to knock some socks off and open some wallets this winter!

DESIGN We've been fleshing out (so to speak) some of the newer areas: Rouen now has roaming bands of Beast Leeches, which is much more exciting than the farmers in the original, and Yonville has completed its transformation into an undead-infested ghost town. Perfect—for a bloodbath!

Also, our big name Hollywood writer has been starting work on the cinematic script, and all I can say is wow! He's a true talent—I think I've mentioned this before, but his screenplay, "Sewer Man: Man from the Sewer," made it to the quarterfinals of the First Annual West Covina Scriptwriting Contest a few

years back. He's also got a pretty well-known blog. I was discussing our concept with him and he really got into the idea of re-imagining Emma as a flawed heroine with a dark past—someone who will stop at nothing to get revenge. This is where we really differentiate our game from so many others.

ART On the art front, the concept guys have been turning out some amazing stuff. I don't know if all of you have had the chance to see the new digital painting of Charles Bovary—just look for the thirty-foot tall beast dripping with slime and made out of the corpses of farm animals. Holy moley, is it sweet! Also be sure to check out the twelve-headed hydra thing (I think its name is Leon) and the disgustingly detailed tentacles on Rodolphe. We also got in brand-new models for the chaingun, the minigun, and the automatic grenade launcher from our outsourcing partners—they're really shiny!

And be sure to stop by the cinematics department for a sneak peek at the

opening movie, too. This is being rendered for us by a fantastic, genuine Hollywood special effects company. I don't want to spoil too much, but let's just say these guys are real pros: they even do that radial blur effect when the monsters roar at the camera, to make it that much more powerful-feeling! Also that wicked awesome speed up/slow down thing when the action gets intense. You can never get enough of that!

AUDIO That reminds me, our composer has just come back from Europe where he recorded the Philharmonic Boys Choir of Prague—I've only heard an early version, but it sounds incredible, really epic. It was just like that music in *Star Wars: Episode One: The Phantom Menace*. You know, the piece that goes "da-da-dadada, da-da-dadada" over and over again? Very sweet, and I can't wait to hear it once it's finished and he's laid the power guitar riffs on top.

Casting continues apace as well. We've been spending a lot of time looking for just the right porn star to be Emma Bovary. We don't

want just any porn star, of course—so we've been asking them all if they play video games. So far they've all said yes, so I think the next step will be to ask them what their favorite games are. I certainly wouldn't want a porn star who only plays *BEJEWELED* to be in our game and I think you all agree.

MARKETING I've been working closely with marketing to make sure they really get the concept behind the game so they can sell our work. They really keyed off the no-holds-barred, edgy tack we've been taking with the material, and have been working on tactics designed to emphasize the dark, gritty world of nineteenth century France and the visceral feel of Madame Bovary's deadly arsenal.

They're putting the finishing touches on a new campaign tentatively titled "Cheat on Your Spouse to Win," where submitting proof of adultery nets you the chance to win a swank Madame Bovary t-shirt, and another centering around racking up life-crushing amounts of debt.

Tying in with that, we've also been working an ultra-rare limited "Entitlement Edition" of the game. The MSRP isn't decided yet but we're aiming for something reasonable that gamers can afford—we're thinking in the realm of \$469.99—which will come in a hand-finished PVC box shaped like a carriage. It'll include an art book, a making-of DVD and some very real-looking "arsenic pills" in a fancy bottle so that our most hardcore fans can "commit suicide!"

KEEP ON ROCKIN'

» Well, that about sums it up. Our game is seriously poised to turn some heads and rack up the sales when it hits store shelves. Let's take a moment to pat ourselves on the back for the great work we've done so far—and join me in thanking our parent company for having the courage and vision to take some risk on a brand-new IP! 🎮

MATTHEW WASTELAND

writes about games and game development at his blog, *Magical Wasteland* (www.magicalwasteland.com).

Autodesk Games Insight

The latest scoop from Autodesk Media & Entertainment



©2009 Ghostbusters, Terminal Reality. ©2009 Atari. Images courtesy of Terminal Reality

“The thing we really liked about Maya was the flexibility of the tools and being able to do things with mouse scripting on the fly. The architecture that’s built into Maya, to me, it’s second to none.”

— Angel Gonzalez
Lead Character and Animation Artist
Terminal Reality

Ghostbusters Redux

Gamers better keep their moist towelettes handy and prepare to be slimed! In Atari’s *Ghostbusters: The Video Game*, based on the 1984 action comedy, gamers must face the marshmallow madness of Mr. Stay Puft and other tortured souls as they navigate spook-filled Manhattan as a junior member of the Ghostbusters crew. The game is a hit and its developer, Dallas-based Terminal Reality, can take a large share of the credit.

It was a mammoth undertaking that spanned almost three years and involved a team of 65 developers, designers, and digital artists. Terminal Reality relied on Autodesk® 3ds Max® and Autodesk® Maya® software, as well as their own custom-made development platform called Infernal Engine, to create a gaming “sequel” to the first two Ghostbusters™ movies.

The Challenge: Make it Authentic

“There had been Ghostbusters games previously, but they were of varying quality and success,” says Drew Haworth, Creative Director. “Our objective was to give players an authentic experience for the very first time, to experience what it would be like to actually use the capture stream to capture and trap the ghosts.”

“You really had to get the characters and their likenesses down,” agreed Adam Norton, Art Director. “Getting the dialogue and the timing and the comedic reactions between the Ghostbusters team was a pretty difficult task.”

The Solutions

Terminal Reality’s concept artist got the ball rolling in late 1996 by putting together a series of high end inspirational drawings that the team reviewed as their baseline environment. “From that point on,” explained Norton, “we used 3ds Max. We then put together what we call a grey box level to simply block out shapes and prototype assets. This allowed us to get it all into the engine very quickly. The design team could then go on to populate the levels with characters.”

“One of the biggest challenges was taking 25 year-old characters and making them seem life-like again,” said Haworth. “If you do a game using intellectual property and you can get the actors that appeared in that movie, you can grab some reference images from those guys. But we couldn’t do that in this case. So we ended up going to not only that movie, but other movies from that time period and just pulled out as many images as possible. The challenge was, we didn’t ever get a really good side profile or a straight on head shot, making it difficult to get the likeness we wanted.”

“So what we did with Maya was create a generic male head rig, which we called ‘Adam’. From that we were able to put multiple cameras based on the camera angle on the image that we were using. We could have up to 10 cameras for the model to reference from any given angle and sculpt his head based on that camera angle. It worked out really well because we were able to get pretty good likenesses overall with every character.”

“We were able to use the Maya software scripting capabilities and tools to get us a really good robust head sculpting system. We ended up using one head for literally every male character in the game.”

The Result



Terminal Reality relied on the scripting capabilities and tools in Maya to develop an innovative head sculpting system for re-creating the life-likeness of the original characters in Ghostbusters.

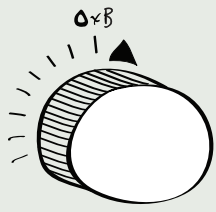
Since the release of the game, reviews from gamers and critics alike have been enthusiastic. “At the beginning, we weren’t sure if would be making a game that was legitimately funny,” says Haworth. “We didn’t know if that would work, we weren’t sure if it was achievable. And that’s something we put a lot of effort into. And from most of what we heard back from players and critics, people actually laugh when they’re playing the game, which is a huge step for us. We think there is a place for comedy in action games, and blockbuster games going forward.”

For more information on Autodesk games software and middleware please visit www.autodesk.com/games.

Autodesk®

Autodesk, Maya, and 3ds Max are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product offerings and specifications at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document. © 2009 Autodesk, Inc. All rights reserved.

TURN UP THE




WICKED



AUDIO IN YOUR GAME WITH

Miles 8

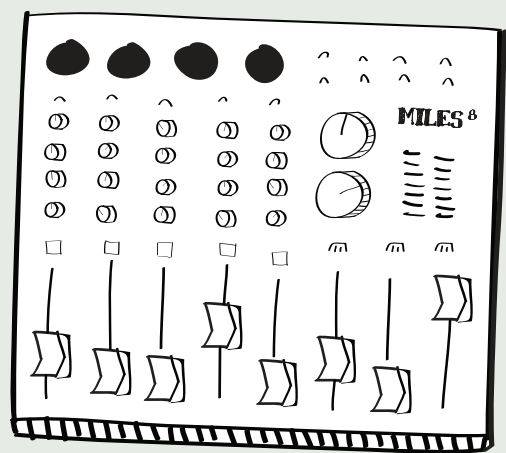
← BRAND NEW!



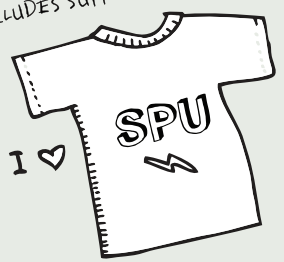
Our multichannel sound system features cool, new

high-level audio tools

+ the world's **FASTEST**



INCLUDES SUPPORT FOR THE PS3



MP3 and Ogg DECODERS.

.....
USING MILES 8 NOT ONLY MAKES YOU

WANT TO TURN IT UP, IT MAKES YOU rad!



www.radgametools.com
(425) 893-4300