UBM

CREATE GAME MAKER AND FLASH GAMES NOW!

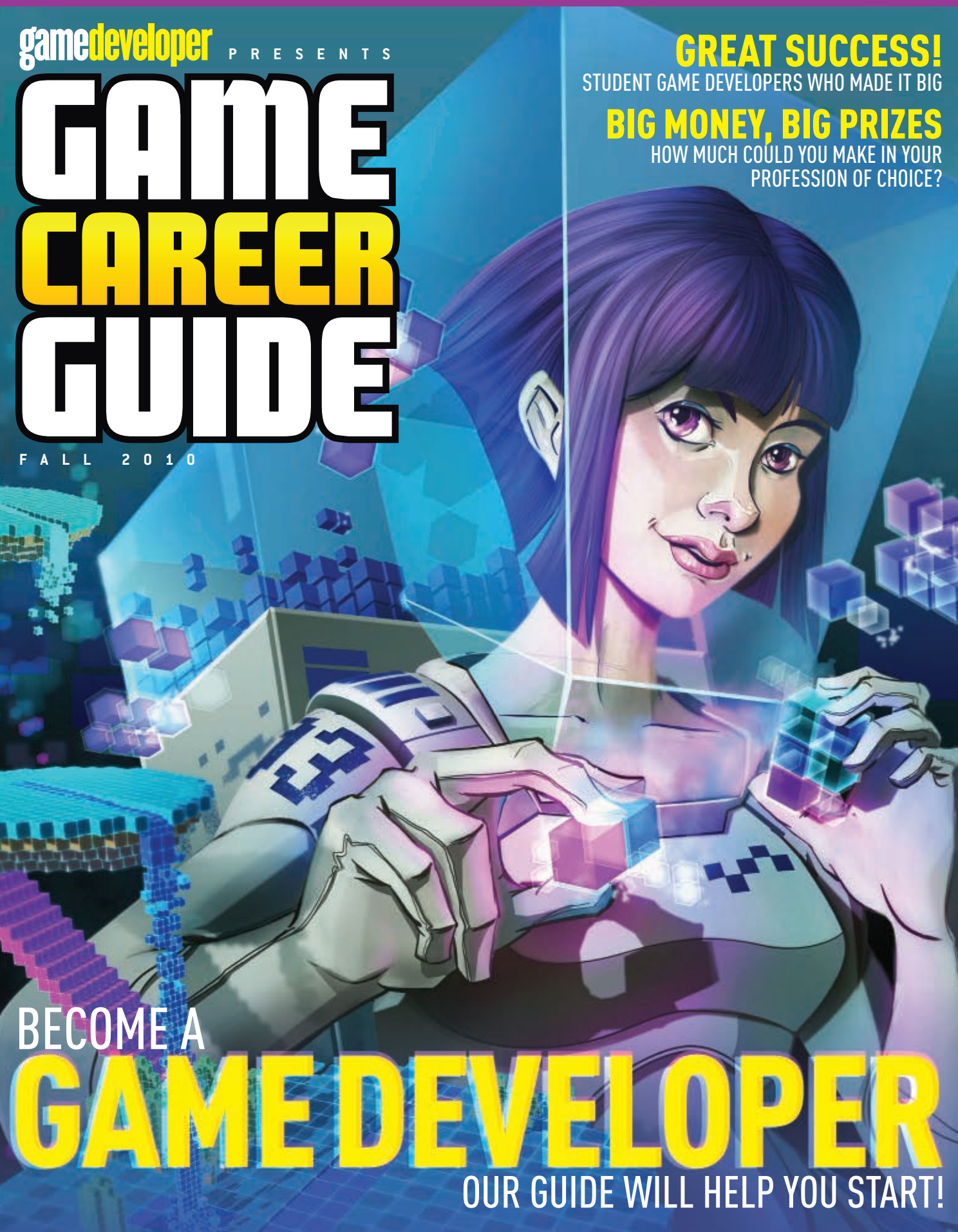
gamedeveloper PRESENTS

GAME CAREER GUIDE

FALL 2010

GREAT SUCCESS!
STUDENT GAME DEVELOPERS WHO MADE IT BIG

BIG MONEY, BIG PRIZES
HOW MUCH COULD YOU MAKE IN YOUR PROFESSION OF CHOICE?



BECOME A GAME DEVELOPER
OUR GUIDE WILL HELP YOU START!

Steps above the rest...

LOCATION
Surrounded by over 150 game companies

SUCCESS
More IGF student awards than any other college or university

REPUTATION
North America's first bachelor degree in game development

DigiPen
INSTITUTE OF TECHNOLOGY

Bachelor of Arts in Game Design
Bachelor of Fine Arts in Production Animation
Bachelor of Science in Game Design
Bachelor of Science in Computer Engineering
Bachelor of Science in Real-Time Interactive Simulation
Master of Science in Computer Science
Part-Time Master of Science in Computer Science

www.digipen.edu

9931 Willows Road | Redmond, WA USA 98052 | Phone (866) 478-5236 | FAX (425) 558-0378 | **www.digipen.edu**

gamedeveloper PRESENTS

GAME CAREER GUIDE

CONTENTS.FALL.2010

STUDENT POSTMORTEM

FEATURES

DEPARTMENTS

COVER ART: VINCENT PEREA

THINK DIFFERENT!

USE YOUR TIME AT SCHOOL TO MAKE SOMETHING UNIQUELY YOURS

IF YOU'RE READING THIS, IT'S LIKELY THAT YOU'VE AT least entertained the idea of attending a school with a game development program, or using your time at school to work toward a career in the game industry.

Assuming you do go to school, more questions emerge. For instance, how should you best use your time? Obviously, you want to make something that will either land you a job or become a viable commercial (or indie, or freeware) product in itself.

If your passion is to work on the next HALO or CALL OF DUTY, you have a straightforward yet incredibly difficult path ahead of you. It's straightforward because it's the most traditional, and thus the simplest to quantify. During your time at school you may want to focus on working in teams, specializing in one area, but diversifying with knowledge of other areas (such as learning a scripting language as a designer, or some programming as an artist).

By the same token, it's difficult because this is the area that everyone will be competing in, from all levels of experience and all other disciplines. It can be hard to differentiate yourself when *everyone* has a couple UE3 mods under their belt, or a decent portfolio of Maya-based school projects. Make sure you go the extra mile, and inject some passion into your work. As THE MISADVENTURES OF P.B. WINTERBOTTOM creator Matt Korba says in our panel interview (pg. 54), "School is a great time to be in a safety net to create something. So, that's what you should do. You should make something. If you want to be an artist, you should make art. If you want to be a designer, you should design things. Nobody is going to give you a job for something you haven't already done. So, you should really use school as a time to do something and show it off."

EXPRESS YOURSELF

>>If you've ever thought of making something a little different, this is when you should do it. School is safe place to try new things, and that extends to game development. In school you can develop games without worrying about your budget (unless you count loans), or turnover, or the whims of a publisher. More importantly perhaps, you can develop games without worrying that they'll sell.

What this means is that you can truly develop whatever you want. If you're really excited about traditional FPS or third-person action games, you can do that, provided you have the skills to match. But if you've any interest in experimental games, or reaching players on an emotional or metaphorical level, or even creating a new genre, school is a great place to do it.

If you look at the Independent Game Festival you'll see that a large percentage of the games in the student showcase are non-traditional, making the best of what resources they have, and striving for something innovative. Developers appreciate new ideas, even if they can't always implement them into their own games.

How should you do this? Korba has more to say on the matter: "I think you just have to do something personal," he says. "Whatever makes you unique and your own personal experience, use that. A lot of times, in student films, people reference other films, and it's becoming the same way in games. It's like people are referencing a lot of other games. But if you take something that's outside of games in your life or your experience and put that in, it's always going to be unique because nobody else has that experience besides you."

Some might argue that more experimental games might not put you in the good graces of a more traditional game developer who's just looking to hire someone to tidy up textures. I submit that these experimental games demonstrate creativity and forward thinking that all developers will appreciate. Not only that, your demo reel or the games you submit to prospective employers may stick out more in their minds simply by virtue of being different.

ITERATE, ITERATE!

>>If you just put in your hours and coast through your classes you're not going to get a very exciting job when you emerge. You've got to go above and beyond. Kellee Santiago of thatgamecompany (FLOWER) says that polish is a good way to rise above. "Certainly, if you look at IndieCade and at the Independent Gaming Festival now, every year, the bar is getting raised on the level of polish in both festival games and also what's coming out of schools," she says. "I think that's a new standard that we're seeing ... as a way of standing out, just having a polished project."

Kim Swift, designer of PORTAL, agrees, saying "... for you to get that polish, do not bite off more than you can chew. It's incredibly tempting to go, 'Oh, I'm going to make this and this and this and this.' No. One idea. Focus. The more time and energy you can put into that one thing and polish the hell out of it will make for a better experience. You'll get that grab in the first five minutes."

Here's the bottom line: When you're at school you have a golden opportunity. You're learning, sure, but you have the right environment and opportunity to really try to differentiate yourself. To do that, you've got to treat your schoolwork like you're already doing what you want to do for a living. A lot of us are used to doing the bare minimum when it comes to school, but if you want to succeed in the game industry, you'll have to prove yourself to the maximum. Get started now!

—Brandon Sheffield

gamedeveloper
www.gdmag.com

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

SUBSCRIPTION SERVICES

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

FOR DIGITAL SUBSCRIPTION INFORMATION
www.gdmag.com/digital

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 | *jmitch@gdmag.com*

PRODUCTION INTERN
Tom Curtis

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**
Brenda Brathwaite **Independent**
Bijan Forutanpour **Sony Online Entertainment**
Mark DeLoura **Google**
Carey Chico **Independent**

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

GLOBAL ACCOUNT MANAGER, EDUCATION
Rafael Vallin e: *rvallin@think-services.com*
t: 415.947.6223

ADVERTISING PRODUCTION

PRODUCTION MANAGER
Pete C. Scibilia e: *peter.scibilia@ubm.com*
t: 516-562-5134

REPRINTS

WRIGHT'S MEDIA
Ryan Pratt e: *rpratt@wrightsreprints.com*
t: 877.652.5295

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
WWW.UBM.COM

around the Arab world there are more than 180 million people under the age of 25*.

yet Arabic game development is still in its infancy.

your opportunity is right here, right now at twofour54º in the heart of Abu Dhabi.

The Arab world is one of the world's fastest growing media markets. With a young population of 180 million under the age of 25, more than 80% with mobile phones*, strong broadband take-up and new gaming innovations, it's a prime opportunity for Arabic gaming businesses.

We empower businesses across all media platforms from production, gaming, digital, animation, broadcast and publishing - with world-class training from **twofour54º tadreeb**, state-of-the-art production facilities with **twofour54º intaj** and supporting Arab creative entrepreneurs with venture funding from **twofour54º ibtikar** to seize the media opportunity the region has to offer.

It's all part of our vision at **twofour54º**, creating a centre of excellence for Arabic content creation in Abu Dhabi.

we are twofour54º. are you?
find us. join us. create with us.
+971 2 401 **2454** **twofour54**.com

twofour54
Abu Dhabi

content creation community

Sources: Arab Media Outlook 2010. Media on the Move 2009. A.T. Kearney. Introduction to Gaming. Michael Moore. Screen Digest. IDC.

One of the many panels at GDC 2010.

PHOTOS BY VINCENT DIAMANTE

Who to Know & What to Do in the Game Industry

JEFFREY FLEMING AND TOM CURTIS

{ EVENTS }

Video game events and trade shows can be a great resource for learning about the industry and game development. Here are a few you should know:

GAME DEVELOPERS CONFERENCE

www.gdconf.com

/// The Game Developers Conference (GDC) began as a small gathering in Chris Crawford's living room, but has since evolved into a weeklong event that hosts thousands of attendees each year. GDC caters to the developers that fuel the industry, and features hundreds of lectures and summits which are divided into tracks for a broad range of disciplines, such as audio, business, design, production, programming, and art. The event also hosts summits that focus on everything from game AI to social games. Among the most popular segments of the event is the Independent Games Festival, where students and independent developers can show off their projects.

Professional and student job seekers alike should check out the Game Career Seminar and visit the Career Pavilion to network with developers, recruiters, and educators. GDC is owned and operated by UBM TechWeb, which also owns *Game Developer* and Gamasutra.com, and will next take place February 28–March 4, 2011 in San Francisco.

Along with the primary GDC, UBM TechWeb's Game Group organizes several satellite conferences

The show floor of E3 2009.

that offer their own set of lectures and focus on various aspects of game design based on their region.

GDC ONLINE

www.gdconline.com

/// Once known as GDC Austin, GDC Online is newly retooled to focus more heavily on online games, from the iPhone market to social games and MMOs, including both traditional subscription-based and free-to-play models. Of course, the show also offers information and lectures for offline game development as well, with tracks in business and marketing, game design, social networking and community, production, and game programming. There are also summits on game audio, game writing and independent games. GDC Online will be held October 5–8, 2010 in Austin, Texas.

GDC EUROPE

www.gdceurope.com

/// GDC Europe is similar to the main GDC, but with a European focus. This year the event will be held in Cologne, Germany, August 16–18, 2010 and features talks on business and management, game design, production, programming, and visual arts.

GDC CHINA

www.gdcchina.com

/// GDC China, held December 5–7, 2010 in Shanghai, hosts talks on online games, global game development, visual arts, game audio, mobile games, and independent games.

GAMESCOM

www.gamescom-cologne.com

/// Remember to make time after GDC Europe because immediately following the conference, Cologne will be hosting Gamescom 2010 on August 18–22. The event is a massive consumer-facing video game exhibition that will be open to industry professionals as well as the general public.

DEVELOP CONFERENCE

www.develop-conference.com

/// The annual Develop Conference features sessions that cover art, audio, business, coding, design, and production. The conference also hosts the Evolve summit, which focuses on the emergence of new platforms, technologies, and markets. Develop will take place July 13–15, 2010 in Brighton, U.K.

TRIANGLE GAME CONFERENCE

www.trianglegameconference.com

/// Many talented developers call the Raleigh-Durham-Chapel Hill triangle area of North Carolina home, such as Epic and Insomniac Games. The Triangle Game Conference makes use of this concentration of quality developers and hosts sessions on game design, business, technology and programming, and marketing. The conference features an exhibition hall, career fair, and game development information for students. The next Triangle Games Conference will take place in Raleigh, North Carolina, at a yet-to-be-determined period in 2011.

TOJAM

www.tojam.ca/home/default.asp

/// The Toronto Independent Game Development Jam is an annual event where independent developers gather to create games and collaborate with their peers. Developers bring their own machine, tools, and sleeping bag for a three-day marathon of game making. TOJam's goal is for participants to create a small-scale yet quality game over the course of the event. Over the last four years, TOJam has generated more than a hundred complete games. TOJam welcomes students, hobbyists, professionals, and even dedicated asset creators who can help generate visuals and audio for other attendees. TOJam will next be held May 2011, in Toronto, Ontario.

INDIECADE

www.indiecade.com/index.php

/// The Indiecade Festival was first held last year as a showcase for independent games and their developers. Since 2007, Indiecade has been a driving force behind the rise of independent games and has organized exhibitions of independent titles at other events, such as E3, SIGGRAPH, and PAX. The next Indiecade Festival will be held October 8–10, 2010 in Culver City, California.

E3

www.e3expo.com

/// Electronic Entertainment Expo has returned to its former status as the largest and most extravagant U.S.-based event in the games industry. Organized by the ESA, E3 is driven largely by publishers and hardware manufacturers, and allows the industry to show off upcoming releases and build excitement for the ever-competitive fourth quarter. Despite the fact that E3 is open only to professionals, more than 41,000 people attended the event in 2009.

GAMES CONVENTION

www.gamesconvention.com/en

/// The annual Games Convention is one of the largest video game trade shows in Europe. The event is open to the public, and attendees can explore the show floor to play upcoming releases and interact with the convention's many exhibitors. Game creators can sit in on lectures on the many aspects of game development in the event's GC Developer's Conference. Games Convention will be held July 9–11, 2010 in Leipzig, Germany.

PAX 2009.

PENNY ARCADE

www.pennyarcadeexpo.com

/// The biannual Penny Arcade Expo is filled with a distinct love for games and the culture surrounding them. Thousands of attendees gather to play upcoming and classic titles, and compete in events such as the show's signature Omegathon tournament. In the evening, attendees can enjoy concerts by artists that celebrate the culture surrounding video games, such as Jonathan Coulton and MC Frontalot. The show also features dozens of panels with industry professionals and a showcase of top independent games, known as the PAX 10. The primary West Coast event will be held September 3–5, 2010 in Seattle, Washington, while the second annual PAX East will take place in Spring 2011 in Boston, Massachusetts.

D.I.C.E.

www.dicesummit.org

/// The D.I.C.E. (Design, Innovate, Communicate, Entertain) Summit brings together big names in the industry on an annual basis to discuss game industry trends and business opportunities. The show is organized by the Academy of Interactive Arts & Sciences, and features the annual Interactive Achievement Awards ceremony. D.I.C.E. will next occur in 2011 in Las Vegas, Nevada.

TOKYO GAME SHOW

http://gs.cesa.or.jp/english

/// TGS is Japan's largest video game trade show, and features four days of new game previews and builds excitement for upcoming releases during the holiday buying season. While the first two days of the show are exclusively for industry professionals, the show floor is open to the public during the final two days.

The event also features workshops, conferences, and the Computer Entertainment Supplier's Association's (CESA) Japan Game Awards. TGS' Sense of Wonder Night is held for developers to show off in-progress or completed projects that demonstrate new or unusual ideas. The show is co-organized by CESA and Nikkei Business Publications, Inc., and will next be held September 16–19, 2010 at the Makuhari-Messe in Chiba City, Japan.

{ ORGANIZATIONS }

These organizations support and regulate the game industry, and a number of them are open to students and consumers.

IGDA

www.igda.org

The International Game Developer's Association is a non-profit organization that connects and educates game developers throughout the industry. The IGDA has chapters in North America, Asia, and Europe, where members can meet with peers, and the organization sponsors events and parties at developer conferences on an annual basis. The IGDA web site features specialized wikis, special interest groups, papers, articles, and columns that span a wide variety of issues in game development. Content on the web site is free, and annual memberships are available at $30 for students and $48 for professionals. Most recently, the IGDA added health care to its list of benefits.

ESA

www.theesa.com

The Entertainment Software Association represents the interests of U.S. video game publishers through anti-piracy programs, government outreach, market research, and intellectual property protection initiatives. The ESA also runs the E3 Expo and supports several charities through its ESA Foundation.

CESA

www.cesa.or.jp/index.php/en

The Computer Entertainment Supplier's Association is a Japanese organization that promotes the video game industry through events such as the Tokyo Game Show and the CESA Developer's conference. It also manages the Computer Entertainment Rating Association (CERA), the rough equivalent of North America's ESRB. The organization also oversees several committees that address industry concerns, such as technology and intellectual property, market research, distribution, and human resources.

VIDEO GAMES VOTER NETWORK

www.videogamevoters.org

The Video Games Voter Network opposes government regulation of video games. Created by the ESA, it encourages consumers to engage in the democratic process by organizing petitions against legislation efforts that restrict game sales.

ESRB

www.esrb.org

The Entertainment Software Ratings Board, created by the ESA, rates the content of video games released in North America and ensures that the industry follows a common set of advertising guidelines. The ESRB receives support from publishers and retailers alike.

GAME AUDIO NETWORK GUILD

www.audiogang.org

The Game Audio Network Guild was created to give game audio creators a common network through which they can meet their peers, share ideas, and advance the implementation of audio in video games. G.A.N.G. offers several benefits to its members, such as professional events, access to its social network, and the ability to interact with the industry's top professionals. Membership is available in a variety of levels, for both students and professionals.

Some of the market research available at www.theesa.com

AIAS

www.interactive.org

The Academy of Interactive Arts & Sciences is a non-profit organization that promotes the excellence of developers throughout the industry. The AIAS holds the annual Interactive Achievement Awards, which honor the best games in the industry based on a vote from the organization's members. The AIAS also organizes the D.I.C.E. Summit for industry leaders. Those who pay for a membership are granted full voting rights if they have been active in the creative and technical fields for at least two years. Students and business sector professionals gain limited voting rights with a paid membership.

{ ADDITIONAL RESOURCES }

There are several other events, organizations, and web sites that game industry professionals should be familiar with, including:

EVENTS

- MI6
- Casual Connect Seattle
- Microsoft Gamefest
- Montreal International Game Summit
- SIGGRAPH
- Taipei Game Show
- London Games Festival
- GameON: Finance 2.0
- SXSW
- Artificial Intelligence for Interactive Digital Entertainment Conference

ORGANIZATIONS

- Computer Game Artists Association
- Game Developers Association of Australia
- IGDA Student Action SIG
- NPD
- The Entertainment Consumers Association

WEB RESOURCES

- Gamecareerguide.com
- Gamasutra.com
- Gamedev.net
- Gamejobs.com
- Gamedevmap.com
- Gamepolitics.com
- Gamestudies.org
- Gameaudioforum.com
- Indiegames.com
- Mobygames.com
- Tigsource.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 has won Game Developer magazine's Best Engine Front Line Award four times and is also one of the few Hall of Fame inductees.

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; "Gears of War 2" for Xbox 360; and "Gears of War 3" for Xbox 360.

Upcoming Epic Attended Events:

Develop
Brighton, UK
July 13-15, 2010

San Diego Comic-Con
San Diego, CA
July 22-25, 2010

GDC Europe
Cologne, Germany
August 16-18, 2010

Gamescom
Cologne, Germany
August 18-22, 2010

Please email:
mrein@epicgames.com
for appointments.

POWERED BY UNREAL TECHNOLOGY

BLUEHOLE DEVELOPS CUTTING-EDGE MMORPG, *TERA ONLINE*, WITH UNREAL ENGINE 3

The latest highly anticipated, massively multiplayer online role-playing game emerging from Seoul is the fantasy epic, *Tera Online: The Exiled Realm of Arborea*.

Developed by a team of 180 at Bluehole Studio, *Tera Online* stands out from the crowded MMORPG space thanks to its use of Unreal Engine 3 technology as well as the creative force behind this original game.

Bluehole Studio's *Tera Online: The Exiled Realm of Arborea*

Bluehole Studio was formed in March 2007 by serial entrepreneur Byung-Gyu Chang and the producer, lead game designer, lead programmer and art director of NCsoft's *Lineage II*, which also used Unreal Engine technology.

"Unreal Engine 3 is an excellent engine that a large majority of our developers had experience with through previous games," said Sung-joong Lew, lead client programmer at Bluehole Studio.

"The Unreal Engine provides essential features like rendering expression and performance, as well as a variety of high-productivity development tools. In addition, the technology's expandability is excellent. One of the engine's strong points is that features implemented during the development of *Unreal Tournament 3* and *Gears of War 2* have been automatically applied to Unreal Engine 3, allowing developers direct access to Epic's latest technology."

Bluehole's technical art director Shin-hyoung Im said his team utilized Unreal Kismet and Unreal Matinee to create the game's opening cinematics.

"We did our best to express the feel of character skins and various materials used in real costumes by utilizing features such as diffuse, normal, specular and specular power provided by the Unreal Engine's standard phong shader," Im added. "We tuned up the phong shader a little bit to make it express unique color impressions for *Tera Online*."

Like many MMORPGs, *Tera Online* will tap into the creative power of its players. Every playable character can be completely customized to take advantage of Unreal Engine 3 visuals.

"Unreal Engine 3 allowed our technology artists to edit materials freely so that various visual looks could be produced," said Lew.

Similar to the deep and intuitive customization of today's best MMORPGs, *Tera Online* players can select face, hair, voice, clothing, weapons, gear and other options for each race. Additional features like skin color, hair color, tattoos, accessories and other cosmetic modifications are provided through material parameter adjustments.

Unreal Engine 3's modularization helped Bluehole add features needed for combat elements without difficulties. Im said that the engine's well-formed development tools allowed the team to easily leverage additional resources as well.

Bluehole said it has been able to get the most out of Unreal Engine 3 technology thanks in part to the past experience of key team members, as well as help through the Unreal Development Network (UDN) and Epic Games Korea.

While *Tera Online* was created over the past three years in Seoul, the game was designed for the global MMORPG market from the outset, which was one of the reasons Unreal Engine 3 was used.

"The Korean MMO market is very competitive," said Harns Kim, associate producer. "The market anticipates emerging killer titles, with a focus on not just a few new elements but the overall quality of the games. We are developing *Tera Online* to the level where every such need is satisfied. From the early stage of development, we've targeted a global market for this new flagship franchise."

Tera Online is the latest game to watch in a growing list of top-tier MMORPGs that run on UE3 technology.

EPIC GAMES

For UE3 licensing inquiries email:
licensing@epicgames.com

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

WWW.EPICGAMES.COM

OUR STUDENTS CREATE GAMES. WE CREATE GAME DESIGNERS.

Get ready to activate your creativity! Gaming today goes far beyond video games – the only limits are your imagination and your skills! Our students can learn to design everything from alien worlds, futuristic cities and blazing car chases to game simulations and interactive entertainment – or whatever else they can dream up.

Take your skills to the next level. Call: 1.888.704.2111

COLLINS COLLEGE
YOUR PASSION • OUR PROFESSION

Main Campus
4750 South 44th Place
Phoenix, AZ 85040
www.collinscollege.edu

Las Vegas
2495 Village View Dr.
Henderson, NV 89074
www.iadtvegas.com

Seattle
At South Center
645 Andover Park West
Tukwilla, WA 98188
www.iadtseattle.com

International Academy of Design & Technology
You imagine. We can get you there.™

Chicago
One N. State St.
Chicago, IL 60602
www.iadtchicago.edu

Detroit
1850 Research Dr.
Troy, MI 48083
www.iadtdetroit.com

Orlando
6039 South Rio Grande Avenue
Orlando, FL 32809
www.iadt.edu

Tampa/Online
5104 Eisenhower Blvd
Tampa, FL 33634
www.academy.edu
www.online.academy.edu

BROWN COLLEGE

1440 Northland Drive
Mendota Heights, MN 55120
www.browncollege.edu

Programs vary by location.Collins College, International Academy of Design and Technology and Brown College do not guarantee employment or salary. AC-0228 CEC2346611 12/2009

JEFFREY FLEMING

EDUCATED PLAY!

INTERVIEWS WITH GAME-MAKING STUDENTS

In a new series for *Game Developer* magazine, we've been interviewing the makers of student games, discussing their tools, influences, and aspirations. Here, we present five of those interviews, combined into a full feature. Students hail from ENJMIN, DePaul University, the University of Central Florida, the University of California Los Angeles, and UC Berkeley.

AZ66

EXPLORING NEW IDEAS IN GAME CONTROL

AZ66 IS AN INTRIGUING UNREAL MOD CREATED BY STUDENTS AT FRANCE'S ENJMIN (NATIONAL SCHOOL OF VIDEO GAME AND INTERACTIVE MEDIA) THAT COMBINES REAL-TIME DATA GENERATED BY THE PLAYER'S HEART RATE WITH A DEVIOUS "CAPABILITY TEST" THAT HAS NO WIN CONDITION. SET IN A DYSTOPIAN WORLD WHERE CITIZENS ARE GIVEN LOGIC AND REFLEX TESTS TO DETERMINE THEIR USEFULNESS TO SOCIETY, AZ66 EVOKES A SINISTER MOOD AND AN ELEVATED SENSE OF PANIC AS THE PLAYER'S HEART RATE HAS AN INCREASINGLY DISRUPTIVE EFFECT ON THEIR PERCEPTION OF THE GAME. WE TALKED WITH AZ66'S DESIGNER STÉPHANIE MADER TO FIND OUT HOW THE GAME (LITERALLY) GETS UNDER THE PLAYER'S SKIN.

The AZ66 team.

Jeffrey Fleming: Can you tell us a bit about the history of the project?
Stéphanie Mader: *We are students at ENJMIN, the only public institute in Europe to deliver a Master's degree in games and interactive digital media. As our school is focused on the specialties of game design, graphic design, sound design, programming, project managing, and ergonomics, a big part of the program is to teach us teamwork, with responsibilities divided according to each member's specialty. During the first school year, we had to make several video games as a team.*

JF: How did the team get together and divide responsibilities?
SM: *Before starting the project, some members of the AZ66 team had already worked together. So, when we came up with the idea of doing a video game using heart sensors, part of the team reformed and some new members joined, in order to have a full development team of six students from each specialty area.*

JF: How much time did it take to complete AZ66?
SM: *For this project we had three months, but we had classes and exams too. On top of that, each member was working on other projects at the same time. Planning the teamwork was a true puzzle.*

JF: How did you approach AZ66's game design?
SM: *Due to the heart sensors, the game had some concept challenges. It was all about bluffing the player and getting him nervous. To accentuate our game design intention to its maximum, AZ66 includes a bomb that can't be defused, never-ending targets that pop up, and a labyrinth with no exit. Dark and dirty visual art, disturbing sounds, and the constant presence of voices all work to distract the player.*

JF: On the technology side, what was the most challenging ?
SM: *We had to learn to work with Unreal Engine, its architecture, its tools, its pipeline, and its script language. As a result, programming and asset integration took most of our time. At the same time, our programmer had to deal with getting the sensor data into the game, while our usability expert worked on the player's comfort when wearing the sensors. But Unreal Engine was also what made our project achievable!*

JF: I'm curious about the biofeedback element in AZ66. How did you measure heart response in the player and how did you connect it to gameplay?
SM: *We were lucky, because we had strong support from Jérôme Dupire who works at CEDRIC, the CNAM French state institute in computer research and development. Jérôme's team created a new version of their PLUG Project with heart sensors, skin response, and temperature, a total of five sensors connectable to the player. In fact, we only used data coming from the heart sensors, due to time constraints. We used the player's variations in heart rhythm to give real-time feedback on his emotional state.*

But remember, it is all about bluffing! The heart rate data didn't really affect gameplay directly, instead it affected the player's perception of the game. We wanted the player to think that his heart response was important by making it audible during the whole game, so he could hear his loss of control. In response to his heart acceleration, we applied graphics filters like motion blur to the screen. To complete our deception, we added some contextual feedback. For example, in the first room, the graphic and sound of the bomb timer becomes strange. In the same way, the speed of the cube that follows behind the player in the labyrinth is directly connected to his heartbeat.

AZ66 TEAM
Project Manager: Stéphan Froment
Game and Level Designer: Stéphanie Mader
Programmer: Antoine Sarafian
Environment Artist: Delphine Soriano
Sound Designer: Aymeric Schwartz
Ergonomics: Mélanie Ginibre
Voices: Laure Nowak and Aymeric Schwartz

AZ66 http://az66.interaction-project.net
ENJMIN www.enjmin.com
PLUG PROJECT http://interaction3d.fr

the devil's tuning fork

ECHOES IN THE DARKNESS FROM DEPAUL UNIVERSITY'S GAMING ELITES

THE DEVIL'S TUNING FORK IS A VISUALLY ARRESTING GAME BY STUDENTS FROM THE GAME ELITES PROGRAM AT DEPAUL UNIVERSITY IN CHICAGO. TAKING ITS VISUAL CUES FROM M. C. ESCHER, THE DEVIL'S TUNING FORK PRESENTS A NIGHT WORLD OF ILLUSIONARY ARCHITECTURE THAT CAN ONLY BE NAVIGATED WITH THE AID OF REFLECTED SOUND WAVES. THE RESULT IS BOTH UNSETTLING AND COLDLY BEAUTIFUL. WE SPOKE WITH PRODUCER MATT LAZAR, PROJECT LEAD AND TECH LEAD JASON PECHO, AND VISUAL DESIGN LEAD KYLE SULLIVAN TO FIND OUT MORE ABOUT THIS UNIQUE GAMING EXPERIENCE.

Jeffrey Fleming: How were students chosen for the DePaul Game Elites program?

Matt Lazar: *From 40 original applicants, 14 were chosen by the team's faculty advisors: Alex Seropian, Patrick Curry, Bill Muehl, Joe Linhoff, and Scott Roberts. One person was added later.*

JF: The Devil's Tuning Fork team is fairly large. Did that present any organizational challenges?

ML: *It was easily the largest group any of us had ever worked in before. We learned so much about teamwork. Every month we became more organized. We started having multiple meetings each week, established clear, concise milestones and kept notes. A big step was choosing leads to make final decisions. By fall, we had become a much more efficient and disciplined unit.*

The Devil's Tuning Fork.

JF: Six months from start to a finished and nicely polished game is pretty remarkable. How quickly were you able to nail down the final design?

ML: *We did not nail down the final design until the fourth month, leaving us only a couple months to make the actual game. Our first month was spent considering various concepts. We prototyped several ideas and after we settled on the idea of using echolocation for game play, we experimented with the look of the game and how it should play—which took more testing and time. It was a constantly evolving process. One of the lessons we learned was that you can never have enough testing. Another was that moving too quickly from the prototyping stage can only cause problems.*

JF: What aspect of the game's development was the most the most difficult?

ML: *Story became one of the most challenging aspects to figure out. Because the mechanic of echolocation was something we had never seen before, it was hard for us to come up with a storyline to match it. We had plenty of internal debates about what it could be or if we even needed a script. Not until late into development did we finalize the story concept.*

To expedite the process, we created a story strike team to write the dialogue, storyboard scenes and direct the voice-acting sessions. It was rapid development, but story ended up being integral to the final experience.

JF: Can you tell us a little about the QE Engine that Joe Linhoff, your faculty advisor, created?

Jason Pecho: *The QE engine is a light-weight OpenGL engine made for educational purposes that acts as a thin layer between the game and the computer. It has grown steadily since Joe introduced it to DePaul University for student use a couple years ago. The major benefit of the engine was that it allowed us to not worry about interfacing with Windows and instead focus solely on the game itself. The engine also comes with a few tools to help development along. It parses Maya files into a suitable format, for instance. In effect, this made Maya not only the art team's tool of choice but also our level editor. We did have to write a bit of pipeline code, but the engine handled most of the translation from Maya to the game.*

A few of the student programmers were familiar with QE before the project started, but some ramp up time was involved. Even those who had experience with QE were constantly learning new features as the development progressed. In addition, the art team had to be educated on the art pipeline to get their models from Maya into the game. Overall, everything worked out quite well.

JF: How was it working with Alex Seropian? What was the best bit of guidance that he gave the team?

ML: *It was great working with Alex. He was so helpful during the concept and prototype phase. Each team member presented at least one idea for the project and Alex gave feedback not only on what would work, but what needed more detail. The whole process prepared us for the months to come. Plus we spent a month of our summer talking about video game ideas with a principle creator of Halo. Could there be any better way to spend your summer?*

As for his best advice, Alex said that when you are creating a timeline for the development of a game, you should plan to have it done approximately halfway to your final deadline because inevitably a game will take twice as long as you think it will. He was right.

JF: How did you arrive at the look of the game? Did going with a more abstract style make asset creation easier?

Kyle Sullivan: *We put a lot of hard work into making the game as visually polished as possible. Our visual design process involved a lot of experimentation and iteration. Nobody had made a game about echolocation before, so we had to start with concepts, try them out in-game, then go back and fix everything that didn't work. We followed this technique for the entirety of the development process.*

As far as asset creation goes, the abstract style of our game didn't really make anything easier. Because we were working with an untested style, we had to take many different things into consideration to keep it unified. For example, all assets had to adhere to our Italian Renaissance/MC Escher aesthetic. This meant following a large list of design rules. If anything, I'd say that our style made our game concept more accomplishable and more fun to make.

THE DEVIL'S TUNING FORK
www.devilstuningfork.com/index.php

galactic arms race

EVOLVING THE SHOOTER

THE STUDENT-BUILT GAME GALACTIC ARMS RACE IS THE INTRIGUING RESULT OF ONGOING WORK FROM THE EVOLUTIONARY COMPLEXITY RESEARCH GROUP (EPLEX) AT THE UNIVERSITY OF CENTRAL FLORIDA. STARTING WITH THE BASIC IDEA OF AN ONLINE MULTIPLAYER SPACE SHOOTER, GALACTIC ARMS RACE ADDS A UNIQUE WRINKLE TO THE COMPETITIVE FORMULA BY FEATURING PARTICLE-BASED WEAPONS THAT EVOLVE INTO NOVEL CONFIGURATIONS DURING PLAY. WE CONTACTED KENNETH STANLEY, THE TEAM'S FACULTY ADVISOR, TO FIND OUT MORE ABOUT THE AUTOMATIC CONTENT GENERATION DRIVING GALACTIC ARMS RACE.

GALACTIC ARMS RACE.

Jeffrey Fleming: Can you tell me a bit about the team that worked on GALACTIC ARMS RACE (GAR)?

Kenneth Stanley: *The team reflects the origin of the game inside a research group at the University of Central Florida. I supervised the project as its faculty supervisor and my Ph.D. student Erin Hastings took the lead in software development and technology integration. The project required integrating novel AI technology developed for the project into the game. The rest of the team was rounded out by volunteers who were mostly undergraduate students interested in gaining experience working on a game. Overall, the project represents a major volunteer and educational effort driven by people's passions, with little financial support.*

JF: What tools did the team use to create GAR?

KS: *GAR is made in XNA. It also uses NEAT and something called "NEAT Particles," which is a technology developed before GAR to allow NEAT to evolve particle systems.*

JF: What is the idea behind the NEAT algorithm?

KS: *NEAT stands for NeuroEvolution of Augmenting Topologies. I invented NEAT at the University of Texas at Austin when I was a Ph.D. student working with my advisor Risto Miikkulainen. As its name implies, it evolves artificial neural networks, which are kind of like little artificial brains. The innovative aspect of NEAT is that the brains it evolves actually get bigger as evolution progresses, which is what the word "augmenting" means in its name. In simple terms, the implication is that behaviors can become smarter and more complex over time.*

NEAT is the core of the algorithm that evolves the weapons in GAR. Actually, for GAR we introduced a variant of NEAT called cgNEAT, which stands for "content-generating NEAT." Believe it or not, a neural network evolved by cgNEAT drives every particle in every weapon in GAR. So the neural networks are actually controlling the way weapons behave. Because the weapons are evolving through NEAT, their behavior can become more complex and intricate over time.

JF: How does cgNEAT decide which weapon to evolve in GAR and which are dead-ends?

KS: *The way cgNEAT works in GAR is that it tracks which weapons people like by observing which ones are fired the most. Those that are popular become the "parents" of new weapons that are spawned in the galaxy. Thus the question of which weapons evolve is answered by which weapons people like. If people like them, cgNEAT makes new variations of them and spawns them in the galaxy.*

JF: Are the evolved weapons specific to a single instance of the online game or are they part of a persistent world?

KS: *In multiplayer mode, the evolved weapons are stored on the server, so they generally persist as long as the server. In that sense, they are part of a persistent world for each server. So the interesting situation is created in which evolution can continue over months or years.*

JF: Is the neural network very processor intensive?

KS: *In GAR, every single particle from every evolved weapon is controlled by a neural network and even when there are ten people on the screen at the same time all firing different weapons at once over a network, GAR players will not experience any slowdown. So from that perspective, today's CPUs are more than capable of handling many simultaneous neural networks being activated at the same time. Neural networks tend to be compact and require only a few floating-point operations, so they can often be less computationally expensive than more traditional control schemes. However, of course, if neural networks are allowed to grow very large, they can start to be more expensive. Yet that size is well beyond what is needed for the type of control in GAR or many other games.*

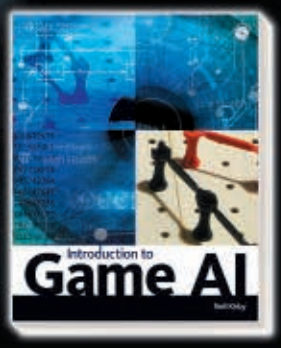
JF: How difficult was it to integrate online play into the game?

KS: *Integrating online play was a challenge because we had to get the system to perform evolution over the Internet, which means that genomes and fitness information literally have to be sent back and forth through messages over the network. There is not much precedent for a real-time Internet-based evolutionary system like that. For example, if a player flies into your view with a weapon you've never seen before, the neural network for that weapon must be transmitted to your computer right away so that the other player's weapon looks the same to you as it does to the other player. However, once the proper information is set up to transmit to the right places, the overall evolutionary algorithm works seamlessly and is not hard to manage.*

GALACTIC ARMS RACE
http://gar.eecs.ucf.edu
EVOLUTIONARY COMPLEXITY RESEARCH GROUP AT UCF
http://eplex.cs.ucf.edu/publications/2009/hastings.ieeetciaig09.html
GOOGLE GROUPS: PROCEDURAL CONTENT GENERATION
http://groups.google.com/group/proceduralcontent
SEARCH-BASED PROCEDURAL CONTENT GENERATION
http://julian.togelius.com/Togelius2010Searchbased.pdf

Your Ultimate Resource

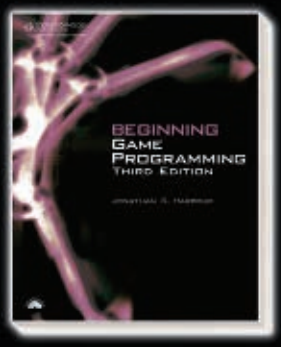
Course Technology PTR is your ultimate game development resource. Our books provide comprehensive coverage of everything from programming, story development, character design, special effects creation, and more, for everyone from beginners to professionals. Written by industry professionals, with the techniques, tips, and tricks you need to take your games from concept to completion.



Introduction to Game AI
1-59863-998-6 • $39.99

iPhone 3D Game Programming All in One
1-4354-5478-2 • $39.99

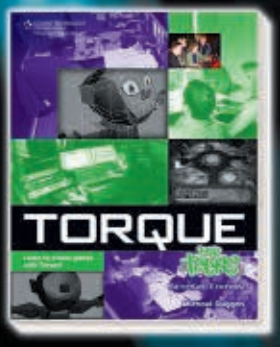
C# Game Programming: For Serious Game Creation
1-4354-5556-8 • $49.99



Beginning Game Programming Third Edition
1-4354-5427-8 • $34.99

Game Programming Gems 8
1-58450-702-0 • $69.99

Video Game Optimization
1-59863-435-6 • $39.99



Torque for Teens, Second Edition
1-4354-5642-4 • $34.99

Advanced 3D Game Programming All in One, Second Edition
1-59863-575-1 • $59.99

Multi-Threaded Game Engine Design
1-4354-5417-0 • $59.99

Mastering Blender Game Engine
1-4354-5662-9 • $44.99

Available at Barnes & Noble, Amazon, and other fine retailers nationwide and online.

COURSE TECHNOLOGY
CENGAGE Learning
Professional • Technical • Reference

courseptr.com

boryokudan rue

GAME NOIR

THE MASS MARKET MAY HAVE PASSED THE ADVENTURE GAME GENRE BY IN RECENT YEARS BUT THE FORM CONTINUES TO ATTRACT A SMALL CADRE OF DESIGNERS AND PLAYERS WHO CRAVE A WELL-TOLD STORY. JOSHUA NUERNBERGER'S 2010 IGF STUDENT SHOWCASE FINALIST BORYOKUDAN RUE IS A NEW GAME WITH A COMPELLING SCI-FI NOIR SETTING THAT REAFFIRMS THE TIMELESS APPEAL OF 2D POINT-AND-CLICK ADVENTURES.

Jeffrey Fleming: How did you like working with Adventure Game Studio to create Boryokudan Rue?

Joshua Nuernberger: *Working with AGS is like working with an old friend. It can pretty much do everything you need it to do as far as point-and-click adventures resembling Monkey Island or any other early '90s adventure games are concerned, and it continues to improve over the years and with a great community to boot. So I've never seen the need to go on to look for other methods of game creation.*

JF: In addition to the AGS forums, have you found other online resources that have been useful to Boryokudan Rue's development?

JN: *Pixeljoint.com is always a favorite source of inspiration, in addition to other art-related sites or blogs. The musings of other game developers—professional or otherwise—such as Ron Gilbert or Ben "Yahtzee" Croshaw also prove useful.*

JF: Was learning AGS' scripting language difficult?

JN: *The learning process was relatively straightforward, and the AGS community is always there if you don't understand something.*

JF: *Have you been able to express all of your ideas with it?*

JN: *Yes, I was able to convey all of my ideas properly, particularly because they didn't rely on the existence of particle systems or physics-based puzzles—minus some handy-dandy box-moving sequences, that is.*

JF: You seem to be pushing at the limits of traditional adventure game mechanics by incorporating action sequences into Boryokudan Rue as well as your earlier game, La Croix Pan. How difficult was it to construct these sequences in AGS?

JN: *I'm by no means a master programmer, so it was basically reiteration off of what I knew at the time, and then construction of a basic prototype during the first couple of weeks of production to make sure that I could actually achieve what I wanted to do.*

In all my games, I like to try and incorporate different aspects of gameplay into the adventure genre, rather than just reusing existing mechanics from the heyday of twenty-or-so years ago.

Wow, was it really that long ago? So that's what led to these different amalgamations of gameplay scenarios.

JF: Has being at UCLA helped you in developing the game?

JN: *I would like to say yes, but for the most part, Boryokudan Rue has been a side project that I've worked on for about two years now, with a lot of the work having been done before I even started at UCLA. Being at UCLA, however, has been a great opportunity to get to know others who are interested in game design as well.*

JF: Boryokudan Rue has a fantastic look to it. The desaturated color palette and rough-edged sketch aesthetic communicates the noir atmosphere really well. What was your process in designing the game's visuals?

JN: *One of my main inspirations in creating the visuals was* Cowboy Bebop. *There's a specific scene set in a gloomy cathedral that really captured the mood I was looking for.*

JF: What art tools did you use?

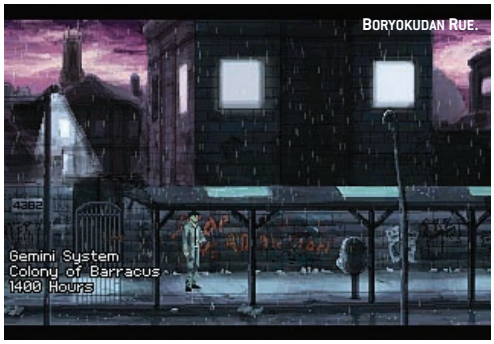
JN: *Tools of the trade include Photoshop, a Wacom tablet, and many, many blank canvases of 320x200 with a pink, beige, and blue palette to jump-start the drawing process.*

JF: Nathan Allen Pinard's moody electronic soundtrack adds an extra dimension to the game.

JN: *Even though we've never met, Nathan has been extremely collaborative with the project. I have been very fortunate to work with Nathan as he's a great composer and his work really adds a lot to the atmosphere.*

JF: A good narrative is crucial in adventure games. Did you write Boryokudan Rue first and then design the gameplay or have they evolved together?

JN: *Like probably most other game narratives, it evolved together as time went on. Creating a narrative for games always depends upon the gameplay, because the story has to set up natural barriers that will reinforce whatever gameplay mechanics you have in mind for that specific game.*



So, writing a story about organized crime naturally sets up an enemy force that would impede the player's progress. In addition, setting a story in a prison-like facility automatically creates some opposition to the goals of the second character, Delta-Six.

JF: What has been the most difficult part in creating Boryokudan Rue?

JN: *Probably the writing. It was a real challenge developing the characters, dealing out the exposition, and just making sure everything in the story "worked." It's hard to tell when things are working or not until you get a secondary opinion on the game. Someone who can specifically tell you "Yes, I get it," or "No, I have no idea what's going on, who that character is supposed to be, or why I'm here." It's just one more reason to playtest your games.*

JF: What is appealing to you about telling a story through games rather than print or film?

JN: *Interactivity. Or that's what I keep telling myself anyway. It's a hard balance to find what is unique to games and exploit that, rather than trying to deceive yourself into thinking that the story you're telling is different just because it's in a game, rather than in a novel or film.*

What's compelling for me in an interactive narrative is the idea of choice, which is a theme that also plays out in Boryokudan Rue. I'd really like to play with choice more in future projects—how does a player's choice or rather, the illusion thereof influence a narrative? Games are different because of their interactivity, so it's important to exploit that in order to provide unique narrative experiences.

BORYOKUDAN RUE TRAILER
www.bigbluecup.com/yabb/index.php?topic=35594.0
Adventure Game Studio
www.adventuregamestudio.co.uk

radiator

WHAT PLAYING OUTSIDE CAN TEACH GAME DESIGNERS

ROBERT YANG'S RADIATOR IS AN ONGOING SERIES OF INTENSELY PERSONAL, HAIKU-LIKE HALF-LIFE 2 MODS THAT CHALLENGE THE TRADITIONAL FPS FORM. IN SUBVERTING THE SHOOTER, YANG SHOWS THAT GAMES CAN PRESENT A FAR WIDER RANGE OF EMOTIONAL EXPERIENCES THAN THEY ARE TYPICALLY GIVEN CREDIT FOR.

RADIATOR.

Jeffrey Fleming: Can you tell me a bit about working with the Source engine?

Robert Yang: *Source is notorious for its convoluted 3D asset pipeline; importing a model from Maya into the engine, for example, is a crazy patchwork of plug-ins, .QC file edits, and batch files. Basic BSP-level sound playback control has been broken since HL1; you need hacks for the simplest function of stopping a sound that's already playing. The prop system is outdated—you can't scale or re-skin anything in the editor. Random entities don't work, the in-editor documentation is really insufficient, and the lack of accurate lighting previews in-editor is glaringly primitive.*

That said, I would never switch engines. Source has the largest mod player-base of any of the major FPS engines. I mean, come on, who keeps UT3 or CRYSIS installed? No one. You'll see a lot of modders complain about the engine, how it won't let them do what you want—well, good design involves designing around constraints. If you want to make a Source mod that won't run on Source, don't blame the engine for it. Build around the problems and modify your design.

JF: It's interesting that you are using a FPS engine to create experiences outside of the shooter genre. What has been the biggest challenge in repurposing Source for the kinds of games that you want to make?

RY: *I don't want to do photorealism, but I'm stuck working roughly within Valve's style because a full non-photorealistic aesthetic would involve re-skinning and modeling new props, changing the weapon models, modifying particle effects, modifying NPC models ... it's a lot of work. Too much work. But I'm confident I'll come up with a style that'll look okay for my future projects and maybe compromise a bit less.*

However, I firmly believe in staying with the first-person perspective for Source mods—that's what it was designed for and that's what it does best. Years ago, I saw an FPS mod for WARCRAFT III; it was really clever how they did it and they clearly put a lot of work into it, but it was really awful by any rational standard. There's still so much unexplored territory in the FPS genre, we should try to map out more of it as modders instead of trying to re-create our favorite genre in something else.

Can you tell me more about your Video Game Level Design course?

RY: *At UC Berkeley, undergraduates can teach their own classes as part of a program called "DeCal." We're the biggest program of our size, with nearly 200 courses offered this spring.*

I taught a DeCal for around three semesters about video game level design. For two semesters, I used COUNTER-STRIKE: SOURCE, but then I got frustrated with students who just ended up building cool-looking stuff without thinking about balance or level flow. The last time I taught it, it was a survey course that covered FPS deathmatch arenas, a TRACKMANIA track, and a STARCRAFT map—and it went really poorly because, again, it's hard to convince the average college student to think about theory instead of building a floating swimming pool or something.

So I decided that computers were the problem, and the best way to teach game design is to play outside. So I've been running a separate "playing outside" course for three semesters that's been an absolute blast. I think all game dev schools should have a "non-digital" component to their game design degrees, ideally a course for outdoor games and a course for board games.

JF: What types of games were being run outdoors? What design lessons did they teach?

RY: *I try to give a wide range of games for them to build a vocabulary with: we play team games that are kind of like leadership/trust exercises, we play capture the flag in libraries, we play scavenger hunts, we play rock-paper-scissors tag ... a lot of good stuff from ludocity.org. Then with that, the students have a common base to design their own outdoor games. So when they get into groups to workshop their own games, they can refer to this canon of sorts—"It's like that zombie game we played, except in teams of three."*

I'd say the main lesson here is about the process of design: Games will have horrible problems that you won't realize until you playtest, and so it's okay to fail—that doesn't mean you're a bad designer. What makes a good designer is someone who is able to fix a broken game design to make it work.

The last unit of the "playing outside" class is always the most difficult because it's about how to embed a worldview into a game. We're familiar with novels and films having ideas and themes, but not games, so it takes a while to learn how to "read games." The example I always use in the class is a student game from the first semester called "Family Dinner."

In "Family Dinner," there are three teams: dogs, kids, and parents. Dogs and kids are allies—dogs circle along the periphery while kids try to sneak them lima beans from the table. Parents serve kids lima beans and try to catch kids in the act of sneaking them to the dogs. What usually ends up happening is the kids start finding loopholes in the rules and parents get really frustrated. Some parents start refusing to serve kids lima beans as a weird form of punishment. Some parents blame other parents for not being watchful enough. It brings out a lot of emotions and dynamics that are embedded in a family dinner—and obviously it represents a cynical worldview of a family dinner, but it's an interesting worldview that only emerges when you play the game.

And that, I think, is the highest form of a game, digital or non-digital: role-playing lets you imagine and understand some larger piece of the real world outside of yourself.

—Jeffrey Fleming

RADIATOR VOL. 1 www.radiator.debacle.us
LUDOCITY http://ludocity.org

Penn
University of Pennsylvania

Penn Engineering's

MASTERS IN COMPUTER GRAPHICS AND GAME TECHNOLOGY

offers recent Engineering and Computer Science grads:

A unique, one-year graduate-level program combining **art** with **engineering** and balancing **theory** with **practice**

A **wide array of interdisciplinary courses** from UPenn's School of Engineering and Applied Science, School of Design, the Wharton School of Business and the Annenberg School of Communication

Access to the **LiveActor Motion Capture Studio** for student game projects

Cutting edge research opportunities at the **Center for Human Modeling and Simulation**

A world class education at University of Pennsylvania's Philadelphia-based **Ivy League campus**

GET IN THE GAME

www.cis.upenn.edu/grad/cggt/

NINTH ANNUAL SALARY SURVEY

GAME DEVELOPER

***GAME DEVELOPER*'S SALARY SURVEY IS THE GAME INDUSTRY'S MOST COMPREHENSIVE LOOK AT** how much money people are making, and how they're making it, broken down by each major discipline. We've run it every year for almost a decade now, and are constantly refining what we report. The main focus is on North America (including Canada), but we also have a special page dealing with the rest of the world.

In this special Career Guide version, we've focused on salaries of three years or fewer, which should be closer to the salaries you might receive upon entering the industry. But you should neither expect nor demand the salaries you're seeing here. Keep in mind that data is subjective to some degree. The folks that have worked for three years will almost certainly have higher pay than those who just started out. Some artists or designers may have started out in QA before getting their current roles, and thus might have hung around in the company long enough to command a higher starting rate. You may not even get a salary when you start—a lot of folks work as independent contractors for QA, art, or other disciplines before they become a salaried employee.

The bottom line is that you should use these pages as a goal, not an expectation. These numbers are more indicative of a salary you might get than something you're guaranteed. If you're not looking to enter the traditional industry, and planning on going it alone as an indie, we've charted that for the first time this year, and you'll find that information on pg. 23.

With all that said, the average game developer salary across all levels and all disciplines last year was $75,573. 2009 was a tough year, with record layoffs, but also record sales. It also saw the rise of social network games, and the proven viability of Apple's App Store. We asked developers several questions to gauge their thoughts about the current state of the industry, and found out that only 13 percent of developers feel there were more jobs in 2009 compared to 2008. A slightly higher group, but still not a majority, felt that the game industry was improving (33 percent).

In spite of all the ups and downs, only 9 percent felt that the game industry was no longer a good place to work, with the grand majority still feeling like this is the industry for them.

Ultimately, if you want to make money, go into business or become a stock broker. If you want to work in the game industry, these numbers should just be icing on the cake.

—Brandon Sheffield and Jeffrey Fleming

NINTH ANNUAL

programmers

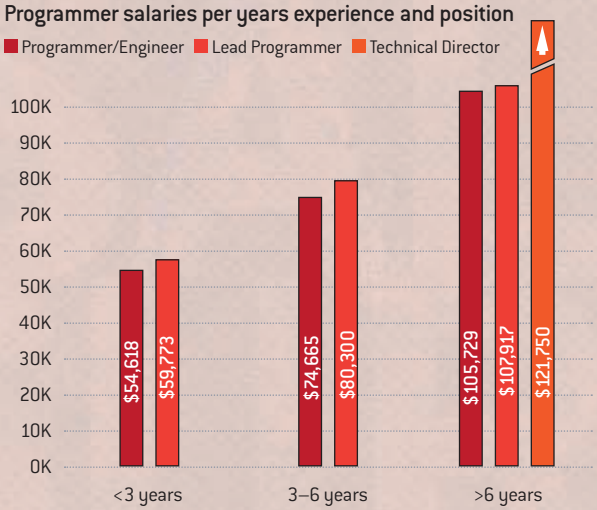
AVERAGE SALARY
3 YEARS OR LESS
$54,975

PROGRAMMING TALENT DRIVES THE GAME INDUSTRY, AND AS A GROUP coders generally enjoy the highest salary among the major disciplines (excepting business). Even so, this year saw a decline in the average compensation, down from last year's $85,024.

Still, the overall pay prospects are good. Junior programmers with three years experience or less years earned an average of $54,975 in 2009 and experienced coders with more than six years on the job got an average salary of $109,567. Inexperienced coders should expect to do a lot of rudimentary systems work as they start out, but if they can hang in there, they will become absolutely invaluable assets to the team.

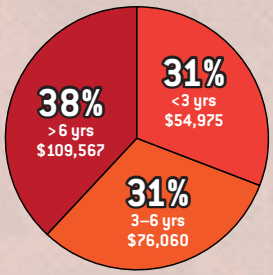
Programmers working in Canada earned an average of $67,937 (USD) in 2009, which is slightly up from last year, while European coders were somewhat down from the previous year, earning $46,198 (USD) on average.

Programmer salaries per years experience and position



ALL PROGRAMMERS AND ENGINEERS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**78%**
Average additional income:	**$15,937**

Type of additional compensation received

Annual bonus	49%
Pension/Employer contribution to Retirement plan	47%
Profit sharing	19%
Project/title bonus	29%
Royalties	16%
Stock options/equity	32%
Percent receiving benefits:	**92%**

Type of benefits received

Medical	98%
Dental	93%
401K/Retirement	80%

GENDER STATS FOR PROGRAMMERS

Gender	Percent Represented	Average Salary
Male	95%	$80,128
Female	5%	$84,062

artists and animators

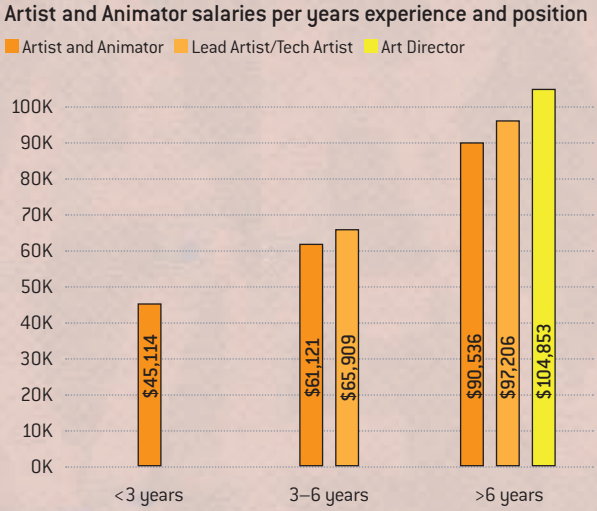
AVERAGE SALARY
3 YEARS OR LESS
$45,200

SALARIES FOR VISUAL ARTISTS WERE UP $1,539 OVER LAST YEAR'S average. Although in general compensation was stable, 14% of the artists surveyed reported a pay increase over the past year, which is the highest percentage out of the creative disciplines.

The bulk of our artists surveyed have been in the business for three years or more, and at the top end their salaries were almost double that of entry level game artists. This likely also means that many low-level artists are on contract, which prospective game industry art hopefuls should be aware of. Looking to the future, the industry definitely values talented art directors and lead/technical artists—across all experience levels these advanced disciplines earned a yearly average of $94,000 and $83,000 respectively.

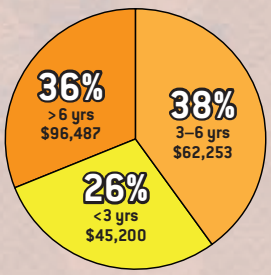
$59,400 (USD) was the mean for Canadian artists, up slightly from last year. European artists brought home a less-impressive $38,152 (USD).

Artist and Animator salaries per years experience and position



ALL ARTISTS AND ANIMATORS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**77%**
Average additional income:	**$12,217**

Type of additional compensation received

Annual bonus	46%
Pension/Employer contribution to Retirement plan	43%
Profit sharing	13%
Project/title bonus	32%
Royalties	17%
Stock options/equity	28%
Percent receiving benefits:	**94%**

Type of benefits received

Medical	99%
Dental	94%
401K/Retirement	81%

GENDER STATS FOR ARTISTS

Gender	Percent Represented	Average Salary
Male	92%	$72,500
Female	8%	$51,071

game designers

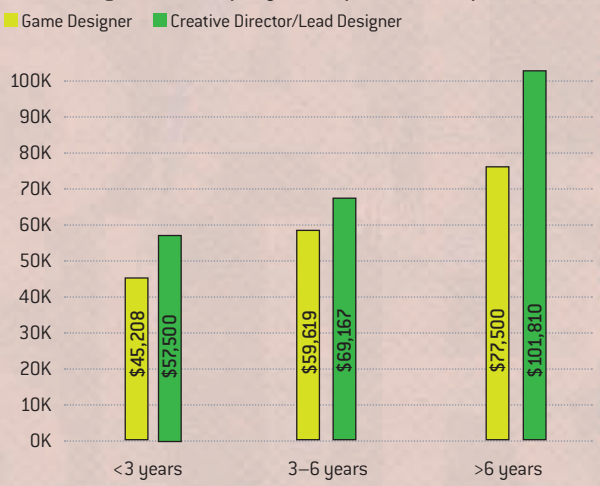
AVERAGE SALARY
3 YEARS OR LESS
$46,019

IN OUR SURVEY, GAME DESIGN ENCOMPASSES A RANGE OF JOBS including creative directors, designers, and writers. The field as a whole enjoyed a boost of almost $2,000 over last year's average salary.

Predictably, creative directors/lead designers were compensated best, earning an average of $90,640 per year, while game designers working in the trenches brought in $61,859 for the year. Game design is a tough field to break into straight off, as it requires a very deep understanding of not only how games play, but how they're made, which many early designers can miss. Learning a scripting language or taking some 3D modeling classes can give you a leg up. Also note that you will likely not become a game writer right out of school.

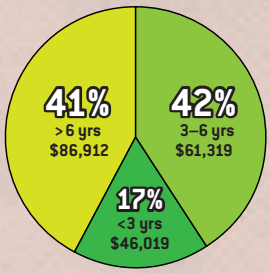
Canadian game designers in general earned $61,520 (USD) this past year. Designers working in Europe reported earning $42,423 (USD), which is up almost $2,000 over last year's survey.

Game Designer salaries per years experience and position



ALL GAME DESIGNERS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**72%**
Average additional income:	**$12,485**
Type of additional compensation received	
Annual bonus	40%
Pension/Employer contribution to Retirement plan	47%
Profit sharing	15%
Project/title bonus	34%
Royalties	12%
Stock options/equity	35%
Percent receiving benefits:	**96%**
Type of benefits received	
Medical	99%
Dental	97%
401K/Retirement	80%

GENDER STATS FOR DESIGNERS

Gender	Percent Represented	Average Salary
Male	92%	$69,790
Female	8%	$62,500

producers

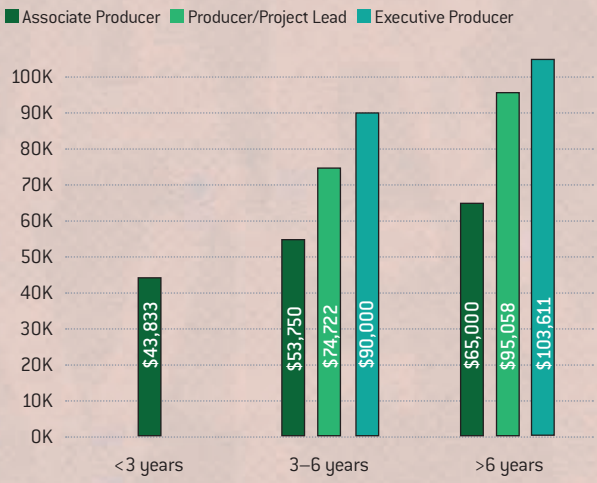
AVERAGE SALARY
3 YEARS OR LESS
$42,000

THE OVERALL SALARY FOR THE PRODUCTION DISCIPLINE IS DOWN $7,823 from last year's average. The majority of our production respondents had three or more years of experience and 48.8% had been working in the industry for six or more years—the highest percentage out of all disciplines. Some respondents may have been counting their entire tenure in the industry though, as many producers are groomed up from other departments. Associate producer has traditionally been a good entry-level role for lateral industry moves, from QA or game journalism especially.

Of the creative disciplines production had the highest percentage of female respondents. According to the data though, women were paid an average of $4,814 less than their male coworkers in the production field.

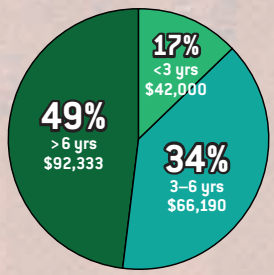
In Canada, producers brought in an average of $87,130 (USD) for 2009 and European producers made an average of $52,125 (USD) for the year.

Producer salaries per years experience and position



ALL PRODUCERS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**77%**
Average additional income:	**$14,565**
Type of additional compensation received	
Annual bonus	53%
Pension/Employer contribution to Retirement plan	42%
Profit sharing	12%
Project/title bonus	26%
Royalties	11%
Stock options/equity	34%
Percent receiving benefits:	**96%**
Type of benefits received	
Medical	97%
Dental	93%
401K/Retirement	81%

GENDER STATS FOR PRODUCERS

Gender	Percent Represented	Average Salary
Male	82%	$75,950
Female	18%	$71,136

NINTH ANNUAL

audio developers

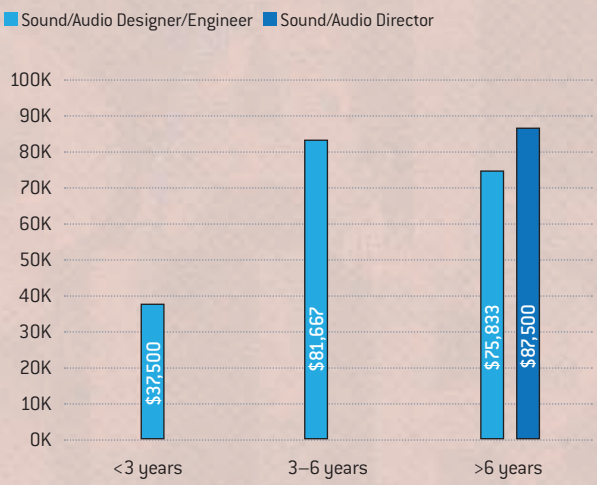
AVERAGE SALARY
3 YEARS OR LESS
$59,643

THOSE WORKING WITH AUDIO, INCLUDING DIRECTORS, COMPOSERS, AND designers, reported earning $3,878 more per year on average than they did last survey. Interestingly, while audio developers were among the least likely to receive additional compensation (excepting QA) for their work, the average additional income for those that did was among the highest ($15,875) in the creative disciplines, particularly with regard to royalties.

Keep in mind that our sample size for the audio department was low, as there are very few in-house salaried audio folks in the industry. Audio is much more like the Hollywood model, where the right person is hired on for the right project, moving on after the project is complete.

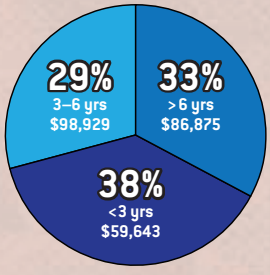
Canadian audio developers earned on average $61,250 (USD) up from last year's $58,929 (USD). Europeans reported a yearly average of $40,833 (USD), down a bit from the previous year's $42,955 (USD).

Audio Developer salaries per years experience and position



ALL AUDIO DEVELOPERS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**71%**
Average additional income:	**$15,875**

Type of additional compensation received

Annual bonus	53%
Pension/Employer contribution to Retirement plan	47%
Profit sharing	20%
Project/title bonus	40%
Royalties	27%
Stock options/equity	27%

GENDER STATS FOR AUDIO DEVELOPERS

Gender	Percent Represented	Average Salary
Male	88%	$81,184
Female	12%	$87,500

Percent receiving benefits:	**86%**

Type of benefits received

Medical	100%
Dental	89%
401K/Retirement	83%

qa testers

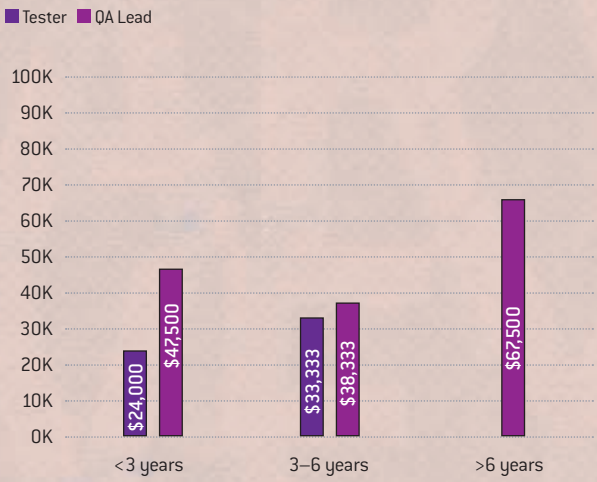
AVERAGE SALARY
3 YEARS OR LESS
$30,714

QUALITY ASSURANCE HAS THE LOWEST BARRIER TO ENTRY OF ALL THE disciplines and accordingly has the lowest pay and least benefits. As a result, turnover is high and the position is often seen as a stepping stone to other areas of game development rather than a career itself. This is a shame, considering how important QA is to the ultimate player experience, and we can only hope more companies start compensating their QA professionals a bit better to keep them around.

The QA discipline, which includes entry-level testers as well as more experienced leads, averaged a salary of $37,905 overall. Testers with less than three years experience on the job reported earning an average of $24,000 annually, while experienced leads that have been in the business for six or more years reported earning $67,500 on average.

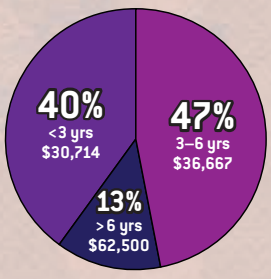
QA workers based in Canada earned an average of $39,375 (USD) per year and those working in Europe brought in $29,500 (USD).

QA Tester salaries per years experience and position



ALL QA TESTERS

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income:	**59%**
Average additional income:	**$5,841**

Type of additional compensation received

Annual bonus	45%
Pension/Employer contribution to Retirement plan	55%
Profit sharing	10%
Project/title bonus	20%
Royalties	0%
Stock options/equity	20%

GENDER STATS FOR QA TESTERS

Gender	Percent Represented	Average Salary
Male	89%	$37,803
Female	11%	$38,750

Percent receiving benefits:	**79%**

Type of benefits received

Medical	100%
Dental	90%
401K/Retirement	77%

Hire education

Our 142 alumni are working at 21 companies around the world making some of the best games and simulations on the market.

These grads are working on everything from the latest Call of Duty and Rock Band to the next Madden Football. And the list keeps growing. So if you are looking to move from playing games to making them, FIEA's curriculum and faculty can teach you how to become a successful programmer, producer or artist.

And since it only takes 16 months to get your master's degree, the first day of your dream job could be sooner than you think.

Florida Interactive Entertainment Academy • www.fiea.ucf.edu • 407.823.2121

FIEA with active learning

UCF Center for Emerging Media

business and legal people

AVERAGE SALARY
3 YEARS OR LESS
$65,833

OUR SURVEY OF THE BUSINESS AND LEGAL DISCIPLINE LOOKED AT a wide range of positions including chief executives and executive managers, community managers, marketing, legal, human resources, IT, content acquisition and licensing, and general administration staff.

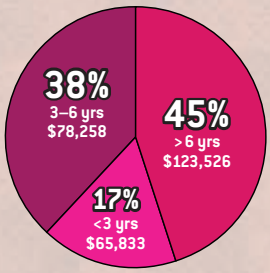
Overall, the average salary for the business sector was $96,408, down $5,735 from last year. Still, those working on the business side are the most likely (89%) to receive additional compensation and enjoy the highest average amount of additional income ($17,807). They are also some of the more experienced workers in the game industry with only 16.9% having been on the job less than three years.

Examining the various job titles revealed considerable variation in yearly salaries. Executive managers earned quite a bit more than the mean, bringing in an average of $129,167 annually. Marketing and PR positions averaged $83,804 and human resources earned $71,136.

Looking at the salary for business-oriented personnel in general for Canada revealed an average of $58,929 (USD) while Europe averaged $59,231 (USD).

ALL BUSINESS AND LEGAL PEOPLE

YEARS EXPERIENCE IN THE INDUSTRY



Percent receiving additional income: 89%

Average additional income: $17,807

Type of additional compensation received

Annual bonus	64%
Pension/Employer contribution to Retirement plan	30%
Profit sharing	29%
Project/title bonus	18%
Royalties	11%
Stock options/equity	41%

GENDER STATS FOR BUSINESSPEOPLE

Gender	Percent Represented	Average Salary
Male	75%	$100,192
Female	25%	$85,227

Percent receiving benefits: 91%

Type of benefits received

Medical	97%
Dental	91%
401K/Retirement	78%

LAYOFFS

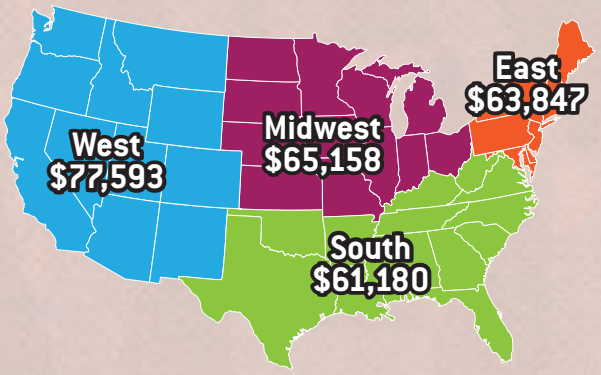
FOR THE SECOND YEAR, WE ASKED OUR SURVEYED DEVELOPERS whether they lost their jobs at any time during the last year. The number is up from the previous period. In 2008, 12% of those surveyed lost their jobs. In 2009, that number climbed to 19%.

Of those who were laid off, 46% found new employment at a game studio or publisher, 17% moved into contract or consulting work, 10% founded a new company, and 16% went into indie development. Unfortunately, 25% were unable to find new work in the game industry.

One particularly interesting statistic is the rather large number of people moving to independent development. Last year, consulting and indie moves combined comprised about 24% of what laid-off developers did with their time. This year, that combined number climbed to 33%, indicating the increased viability of independent game development or working on a game-by-game basis.

AVERAGE SALARY BY U.S. REGION

(across all levels of experience and disciplines)



TOP 5 STATES WITH HIGHEST AVERAGE SALARIES

(across all levels of experience, excluding states with low sample size)

		AVERAGE SALARY	PERCENT WHO OWN HOMES	AVG. SALARY OF HOMEOWNERS
1	California	$80,557	31%	$107,370
2	Washington	$73,981	41%	$93,728
3	New York	$64,167	20%	$77,500
4	Florida	$60,357	38%	$70,395
5	Texas	$60,326	43%	$70,245

AVERAGE SALARY BY U.S. REGION BY DISCIPLINE

	EAST	MIDWEST	SOUTH	WEST
Programmer	$63,198	$71,154	$64,542	$91,805
Art and Animation	$62,500	$54,265	$60,521	$80,196
Game Design	$67,262	$75,682	$56,842	$73,840
Production	$68,250	$41,389	$71,167	$81,474
Audio	$70,833	$87,500	$50,000	$98,000
QA	$36,250	—	$28,214	$41,591
Business	$87,885	$122,500	$89,868	$99,265

AVERAGE SALARY FOR HOMEOWNERS VS. NON-HOMEOWNERS BY U.S. REGION

	EAST	MIDWEST	SOUTH	WEST
Homeowners	$82,500	$82,244	$73,776	$99,108
Non-Homeowners	$52,266	$47,361	$49,493	$63,717

AVERAGE SALARIES IN THE U.S., CANADA, AND EUROPE

(across all levels of experience, by discipline, given in USD)

	U.S.	CANADA	EUROPE
Art and Animation	$71,071	$59,400	$38,152
Programmer	$80,320	$67,937	$46,198
Game Design	$69,266	$61,520	$42,423
Audio	$82,045	$61,250	$40,833
Production	$75,082	$87,130	$52,125
QA	$37,905	$39,375	$29,500
Business	$96,408	$58,929	$59,231

SALARY SURVEY

AVERAGE SALARY BY EDUCATION LEVEL AND DISCIPLINE

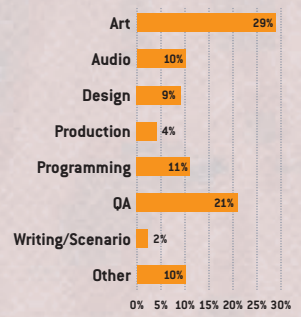
(across all levels of experience)

	ART	PROGRAMMING	DESIGN	AUDIO	PRODUCTION	QA	BUSINESS
High School Diploma or GED	$74,167	$79,375	$56,136	$114,167	-	$35,000	$72,500
Some College	$101,310	$98,500	$79,929	$140,000	$78,833	$28,056	$75,833
Technical Certification	$52,500	$79,167	$80,000	$87,500	$52,500	-	$82,500
Associates Degree	$69,000	$68,409	$67,500	$60,000	$60,833	$40,000	$65,000
Bachelors Degree	$66,492	$76,383	$67,767	$72,045	$70,643	$40,119	$96,429
Some Graduate	$93,500	$91,923	$68,611	$117,500	$91,875	$30,000	$110,500
Masters Degree	$68,816	$78,013	$63,750	$32,500	$88,125	$97,500	$102,976
Some Doctoral	-	$111,786	$57,500	-	$127,500	-	-
Doctoral Degree	-	$77,500	-	-	-	-	$110,833
Some Post-Doctoral	-	-	-	-	$107,500	-	-
Post-Doctoral Degree	-	-	-	-	-	-	$167,500

THE INDIE REPORT

The Behemoth does extra promotion on the side by making vinyl toys.

CONTRACTORS BY JOB FUNCTION



AS A NEW PART OF OUR SURVEY, WE HAVE INCLUDED new data on independent developers and independent contractors (meaning those who are not part of a contract development team, such as an art outsourcer).

Of those who match the above description, 40% identified as an independent contractor, 30.4% were members of an indie team, and 29.6% said they were an individual indie developer.

INDIES BY JOB FUNCTION

Art	48%
Audio	26%
Design	75%
Production	53%
Programming	66%
QA	50%
Sound	26%

So how much did these people make? Independent contractors fared the best, with an average compensation of $45,137. Among full-on team-based indie developers, the average was less than half, at $20,248. Individual indie developers fared the worst, making a scant $11,638.

Not all indie developers make their money exclusively through game development though. To that end, we also inquired about revenue from related merchandise, such as t-shirts, comics, toys, and the like. Less than 10% of our indie respondents made extra money through those means, and of those, 23% made less than $100 on their extra promotions. 21% made between $100 and $1,000, 21% made between $1,000 and $5,000. 18% made between $5,000 and $20,000, and a rather surprising 16% of indies involved with non-game merchandising made over $20,000 on their extra ventures.

We also asked the entire group, both indies and contractors, whether they had ever worked at a traditional salary-based game developer, and a very large number had not—68%, in fact. This is significant especially when you consider the fact that almost 27% of our respondents identified as a non-salaried game developer. That 68% of that larger number have not worked at a traditional game developer before indicates that the roads to game development are clearing again, and that the traditional story of having to work up from QA perhaps applies less than it did in the past.

JOB FUNCTIONS

Indies are often necessarily a one-stop shop for all aspects of game development, so rather than ask them what one element of game development they were responsible for, we asked them to check all boxes that apply. So read the indie chart as "75% of indies worked on at least design." In the case of contractors, we asked them to choose only one area in which they made the majority of their money.

PHOTOGRAPH BY TOPHER CANTLER, DESTRUCTOID.COM

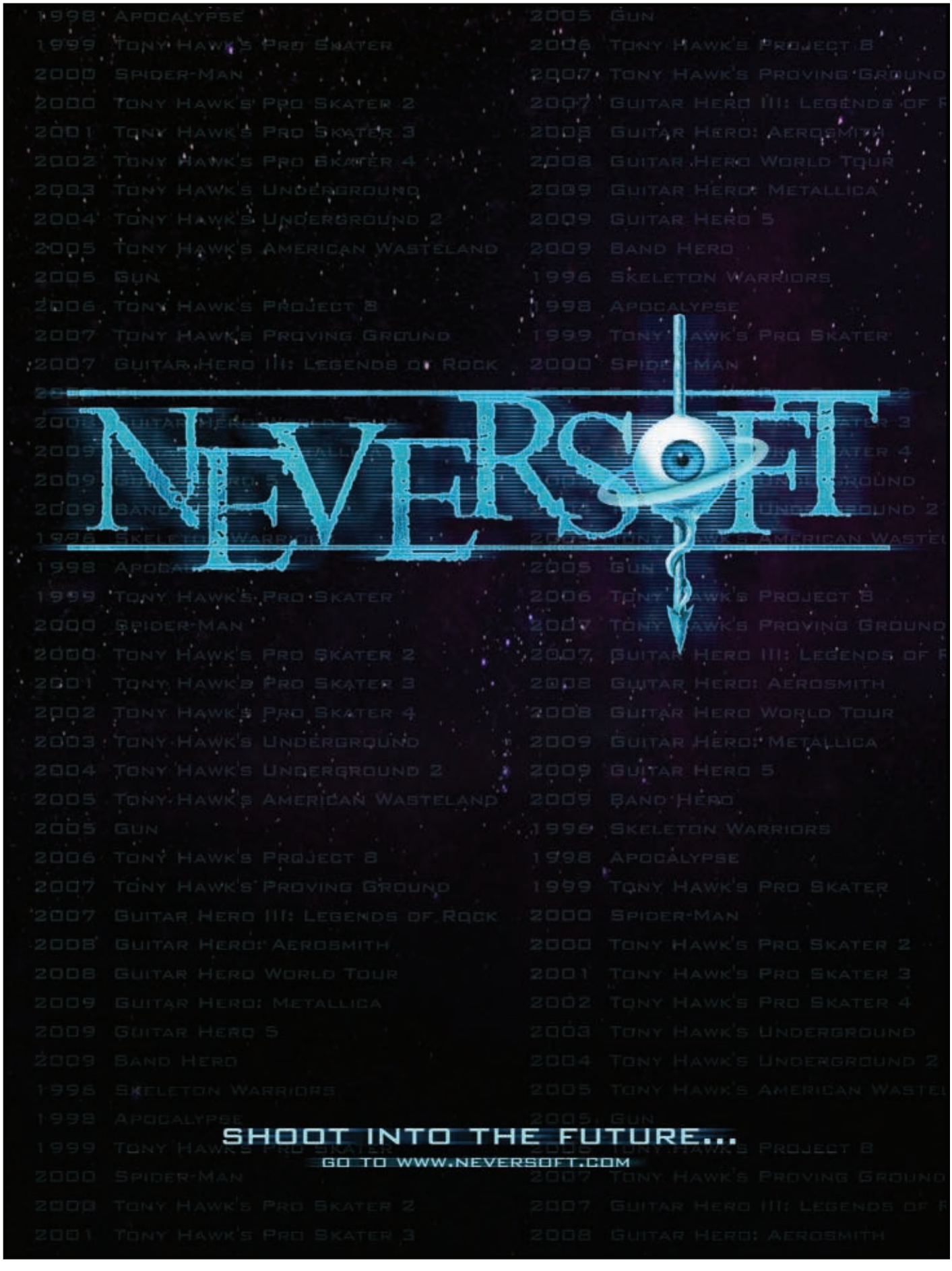
METHODOLOGY

NOW IN ITS NINTH YEAR, the *Game Developer* Salary Survey was conducted in February 2010 for the fiscal year January 1, 2009 through December 31, 2009, with the assistance of Audience Insights. Email invitations were sent to *Game Developer* subscribers, Game Developers Conference attendees, and Gamasutra.com members asking them to participate in the annual survey.

We gathered 4,050 responses from developers worldwide but not all who participated in the survey provided enough compensation information to be included in the final report. We also excluded salaries less than $10,000 and the salaries of students and educators. The small number of reported salaries greater than $202,500 were excluded to prevent their high numbers from unnaturally skewing the averages. We also excluded records that were missing key demographic and classification numbers.

The survey primarily includes U.S. compensation but consolidated figures from Canada and Europe were included separately. The usable sample reflected among salaried employees in the U.S. was 1,014, for Canada 275, and for Europe 378; and 605 for indies and independent contractors who provided compensation information worldwide.

The sample represented in our salary survey can be projected to the U.S. game developer community with a margin of error of plus or minus 3.06% at a 95% confidence level. The margin of error for salaried employees in Canada is plus or minus 5.9%, and is 5.0% for Europe.


1998 Apocalypse
1999 Tony Hawk's Pro Skater
2000 Spider-Man
2000 Tony Hawk's Pro Skater 2
2001 Tony Hawk's Pro Skater 3
2002 Tony Hawk's Pro Skater 4
2003 Tony Hawk's Underground
2004 Tony Hawk's Underground 2
2005 Tony Hawk's American Wasteland
2005 Gun
2006 Tony Hawk's Project 8
2007 Tony Hawk's Proving Ground
2007 Guitar Hero III: Legends of Rock
2008 Guitar Hero: Aerosmith
2008 Guitar Hero World Tour
2009 Guitar Hero: Metallica
2009 Guitar Hero 5
2009 Band Hero
1996 Skeleton Warriors
NEVERSOFT
SHOOT INTO THE FUTURE...
GO TO WWW.NEVERSOFT.COM

ADAM SALTSMAN

flash*forward*

step-by-step creation of a platform game in adobe's flash

OVER THE LAST FEW YEARS, Flash has become the de facto standard for browser-based games, and not just crappy banner ad games, either. With the help of some pretty amazing developers, Flash games are shedding their "Flash game" stigma and enjoying a bit of a renaissance. Top-selling (not to mention fantastic) console games like CASTLE CRASHERS and WORLD OF GOO both have their roots in the rapid prototyping world of Flash. This tutorial will introduce you to Flash from the programmer's perspective, and walk you through the creation of a simple platformer.

GETTING STARTED

>> Building a game, like building an Ikea desk, requires a few simple tools. Fortunately, they're free, fun to use, and very powerful! Programmers call these tools IDEs, or Integrated Development Environments. If it helps, you can think of them as Microsoft Word for programmers. You type your code, hit save, and then run your game. If we're talking about Flash, we're particularly interested in IDEs that are friendly toward or compatible with ActionScript 3 (AS3), the programming language that powers Flash games.

Windows. For Windows users, we recommend downloading an awesome piece of free software called FlashDevelop from the official web site; just click on the yellow box in the upper right that says "Latest Release." FlashDevelop users need to download something called the Flex SDK from Adobe, which is free as well, from the official Adobe site (see Resources on pg. 30 for all of these URLs). Unzip that big pile of files into your directory of choice; we recommend C:/Flex. When FlashDevelop asks where you installed Flex, point it to this directory. That's it!

Mac. For Mac OSX users, we recommend the Flash Builder trial, which is free for 30 days and includes the Flex SDK (the collection of files and utilities that allow us to program Flash games), so there's just one download and one installer. You can download Flash Builder from the official Adobe site.

Experts. If you've never written a computer program before, we highly recommend using FlashDevelop or Flash Builder, which we just mentioned. But if you have some experience, you might want to directly download the Flex SDK from Adobe, and then you can program Flash games in your text editor of choice. Flash community mainstay Senocular has an excellent tutorial on how to get started with MXMLC (also in Resources), the free compiler included in the Flex SDK. The rest of this article applies regardless of which method you're using.

For anyone new to creating games, using an established game library can help speed things up by taking care of a lot of the basics of game code. There are two libraries we recommend; both are free and cover the fundamentals of 2D games, including sprites, tilemaps, collisions, and scrolling. These libraries are Flixel, and FlashPunk. In this article, we'll focus on Flixel, because I created it myself.

If upon completing (or failing to complete) the tutorial, you feel like Flixel just isn't for you, I heartily encourage you to try FlashPunk. Even though there is no tutorial for FlashPunk in this issue, there are a lot of great online resources for it. However, a lot of new designers have had good luck with Flixel, so we're going to take a short tour through how a Flixel game is put together so things aren't too overwhelming when we eventually get to the nuts and bolts of the tutorial. >>>

WHAT'S IN A GAME?

>> Games have a lot of different stuff in them, but we're going to take this first game pretty slow. We'll start by describing all the various components you'll need. Flixel is an "object-oriented" library or API, which means it is a collection of source code files. Each file represents a different kind of basic game element, including characters (or "sprites"), sound effects, and a helper or "container" for the game itself. Let's start with that last one!

The FlxGame Object. FlxGame is a special object that is only created once, and you only really use it right at the start of the game to set up some important basic properties. Other than that, it acts like both a container and manager. FlxGame is in charge of keeping track of your game objects and making sure they get updated and drawn, so it does most of its work behind the scenes.

The FlxState Object. To help keep things organized, Flixel uses a simple "state engine." This means that the different parts of your game that have different behavior can be written or scripted in different files. For example, most Flixel games have a "menu" state and a "play" state, so you don't have to muck up your gameplay code with menu display or list controls. All your real game design work and game objects will be done as part of a "state," not part of FlxGame.

The FlxSprite Object. A FlxSprite, or "sprite," is a graphical representation of a game object. It might be the player's avatar, a collectible powerup, a giant boss, or some atmospheric particle effects. By adding sprites to the game's state, you can change their position and draw them to the screen and make a game. The only visual objects that aren't sprites in Flixel are big level objects, like tile maps (below).

The FlxTilemap Object. Since levels are so big, it can be helpful to have a special object to store them and draw them more efficiently. FlxTilemap takes an array of numbers and uses those numbers to draw tiles (little squares of level graphics) to the screen, like in SUPER MARIO BROS. or LEGEND OF ZELDA. It can also use those numbers to determine what objects the character can run into, or collide with, and which tiles should be treated as non-existent. Just like a FlxSprite, FlxTilemaps are first created and then added to the game state.

FlxG And FlxU. These oddly named objects, or classes, are short for "global" functions and "utility" functions, respectively. They contain a bunch of helpful things that we use a lot in game development, like checking for keyboard presses or bumping objects into each other.

LISTING 1

```
package
{
	import org.flixel.*;

	[SWF(width="640", height="480",
	backgroundColor="#000000")]

	public class EZPlatformer extends FlxGame
	{
		public function EZPlatformer()
		{
			super(320,240,PlayState,2);
		}
	}
}
```

LET THE TUTORIAL BEGIN!

>> That's all we're going to use to make our MARIO-esque platform game. To recap, our ingredients include a game object, which contains a state object, which in turn contains sprites and a tilemap. If the game were a Russian nesting doll, FlxGame would be the biggest doll, FlxState would be the medium doll inside, and inside FlxState, there would be a whole pile of tiny dolls. Let's make our biggest doll first!

Step 1: Adding Flixel. In your development environment of choice, create a new AS3 Project (AS3 is our programming language, remember?), and add the Flixel source folder to the project. Depending on what IDE you are using (as discussed in the Windows and Mac sections on pg. 25), you might have to copy the whole Flixel folder right into your new project's source folder on your hard drive. In Flash Builder, you can look up the folder where you downloaded Flixel and add it to your project that way.

Step 2: Creating the Game Object. This is going to be a long step, even though it's a short file we won't use again for this game. Just a heads up! In the new project's main or application file (usually, it has the same name as your project), we need to add the code in Listing 1.

Let's go through this line by line. Since this is the start of the project, we don't want to miss any details quite yet.

```
package
{
```

This is simply telling your Flash IDE that you're creating a file in the default source directory. This may not be ideal if you are sharing source code with another project or something, but for quick Flixel sketches, it's no problem.

```
	import org.flixel.*;
```

This tells the Flash IDE that you want to include all the Flixel source files. If you wanted to include a specific file, you would just us that filename (e.g. "FlxGame.as") instead of "*." Including all the Flixel files is fast and easy and doesn't really have any penalties, so it's usually simplest to just use the asterisk wildcard.

```
	[SWF(width="640", height="480",
	backgroundColor="#000000")]
```

This is a special "pre-processor" command to your Flash IDE that tells it what size to make your Flash game project. We want our sample game to run at a full resolution of 640x480 (more on this later).

```
	public class EZPlatformer extends FlxGame
	{
```

Whoa! What is this? There are a few different things going on here. public refers to the "visibility" of the class. For now, let's not worry about it. Suffice to say that all our stuff for this project will be public. The class part refers to what FlxGame is—programmer-speak for object, basically. Classes can have variables and functions, and even store other classes. EZPlatformer refers to what I named our project. The next part, extends, means we're making a new class, which is called EZPlatformer, that is based on FlxGame, our "largest Russian doll."

```
		public function EZPlatformer()
		{
```

This is what we call a function declaration, and since it has the same name as the object, it's called a constructor declaration. That means

ACADEMY *of* ART UNIVERSITY

FOUNDED IN SAN FRANCISCO 1929 BY ARTISTS FOR ARTISTS

Gaming Artwork by Rob Arnold

Gaming Artwork Collaborative Student Project

Gaming Artwork Collaborative Student Project

TAKE CLASSES ONLINE OR IN SAN FRANCISCO

Advertising
Animation & Visual Effects
Architecture*
Art Education*
Fashion
Fine Art
Game Design
Graphic Design
Illustration
Industrial Design
Interior Architecture & Design
Motion Pictures & Television
Multimedia Communications
Music for Visual Media
Photography
Web Design & New Media

ENROLL NOW

EARN
YOUR AA, BA, BFA, MA, MFA OR M-ARCH ACCREDITED DEGREE

ENGAGE
IN CONTINUING ART EDUCATION COURSES

EXPLORE
PRE-COLLEGE SCHOLARSHIP PROGRAMS

WWW.ACADEMYART.EDU
800.544.2787

79 NEW MONTGOMERY ST, SAN FRANCISCO, CA 94105

Accredited member WASC, NASAD, Council for Interior Design Accreditation (BFA-IAD), NAAB (M-ARCH)

**Degree program not currently available online.*

this function, or set of instructions, is automatically called when the object is created. Since this is our main object, Flash creates it automatically; we don't have to do anything. You'll notice another weird bracket there—each "open" bracket indicates that we're opening a different kind of Russian nesting doll. In this case, what we're telling Flash is that our constructor (`EZPlatformer()`) belongs inside our `EZPlatformer` object, which belongs inside our default source tree package from the first line.

```
                super(320,240,PlayState,2);
            }
        }
}
```

This is maybe the most important line of code in this file—super refers to the object you're extending, in this case FlxGame. Since there is no function attached to it (`super.update()` for example), you know that what we're calling is FlxGame's constructor. Then we give it some important parameters. In order, these parameters indicate the width, height, starting game state, and "zoom" level of your entire game. Notice something funny? The width and height of the game are exactly half the size of the width and height we passed to Flash earlier in this same file. However, "width" (320) times "zoom" (2) exactly equals the number we put up there: 640. Make sense? We're going to display the game at 640x480, but the game itself is only 320x240. We're going to blow it up, or zoom into it, so that each pixel takes up twice as much space. This leads to a kind of chunky, retro aesthetic, but it's also a good way to get extra performance out of your Flash game. Finally, we close our function, then our game object, then our package with closed brackets.

LISTING 2

```
package
{
        import org.flixel.*;

        public class PlayState extends FlxState
        {
                override public function create():void
                {
                        //Player and map setup will go
                        here
                }
        }
}
```

LISTING 3

```
add(level);

//Create player (a red box)
player = new FlxSprite(FlxG.width/2 - 5);
player.createGraphic(10,12,0xffaa1111);
player.maxVelocity.x = 80;
player.maxVelocity.y = 200;
player.acceleration.y = 200;
player.drag.x = player.maxVelocity.x*4;
add(player);
```

Step 3: Creating A Game State. Okay—save that file and close it. We're done with it. On to more exciting things! We probably won't dive as deep into this code, but I'll try and break it up into more manageable steps as well. First, we want to make a new AS3 file called "PlayState," which will store all our gameplay code including creation, gameplay, controls, and drawing. See Listing 2.

Look familiar? It's pretty close to our other class, but this time we're extending "FlxState" instead (the medium Russian doll). This is technically a functional game state, but it won't do a thing because we haven't asked it to yet! So next up we're going to declare a bunch of variables to hold our objects.

Step 4: Adding Some Variables. Not all game objects need to be stored in variables, but if you want to refer to that object later, then you need a way to look that object up. So let's add all the variables for our game:

```
public class PlayState extends FlxState
{
        public var level:FlxTilemap;
        public var player:FlxSprite;

        override public function create():void
        {
```

Here we have variables to store the game's level or environment, and then one variable for the player. Usually, when you're designing a game or a prototype, you won't know ahead of time all the variables you'll need, and you'll add one or two at a time as you think of new features or options. This is a great way to start small and add complexity in a sensible, organic way.

Step 5: Creating the Background. It's nice to put something behind your level, whether it is a blue sky or a creepy cave, or even something abstract like this:

```
override public function create():void
{
        bgColor = 0xffaaaaaa;
```

Here, we are assigning a variable called "bgColor" a weird series of numbers and letters. "bgColor" is a built-in variable, part of the FlxState object, that controls the background color of the game. The default value is black, and we want something brighter, like light gray. In Flash, colors are stored using kind of a weird format. It's similar to the hex values that you might use on a web page, just a little different:

```
WEB:   0xRRGGBB
FLASH: 0xAARRGGBB
```

Here, "A" refers to the alpha of the color. "ff," like we used above, means fully opaque. The other values of, course, add up to light gray.

Step 6: Creating A Level. Now that we have a blank gray background, we see how Flash handles colors. Now we need to create a level for the player to explore. There's a large chunk of code you have to add right below the bgColor code. It's too big to display here, so grab it from www.gdmag.com/resources/code.htm and slap it in there (the file will be called PlatformingLevel). It should look like a big square of text.

Look at all that code! Most of that is an actual 1:1 representation of the game level. It is 40 tiles wide and 30 tiles tall. 1s indicate where a solid wall or floor tile will be placed, while 0s represent blank or open space where the background will show through. Once we create the data

Our game environment. No player yet!

or layout for our map, all we need to do next is create a new map, and add it to the game state. Those two lines in the middle (`level.auto` and `level.loadMap()`) load the level data up in the tilemap object, and tell it we want Flixel to place the tiles for us automatically. Now we can run the game and see our level in action.

Step 7: Adding the Player. Now that we have a level, we can add a player character to run and jump around in there. See Listing 3.

This code creates a new sprite and stores it in "player." It positions the player at the top center of the screen, and then creates a new graphic to represent the player: a 10x12 red box (using that same color format from before). To get smooth, Mario-like movement, we are going to use Flixel's acceleration variables instead of just changing the position or velocity manually. To use the acceleration system, we want to set some barriers or boundaries to the acceleration behavior, so we set the "maxVelocity" on each axis (x is horizontal, y is vertical). Then we permanently set the player's vertical acceleration to 200. This simulates the force of gravity on the player object. To make sure the player slows down when they let go of a direction key, we're going to set the "drag" of the player object to 4 times the maxVelocity. This means if the player is moving left and suddenly stops, they will slide to a halt in about a quarter of a second. This feels smooth but not too slippery in our game!

Step 8: Updating the Player. Now, if you hit "compile," a red box appears, and falls and falls and falls. That's not very useful, is it? It's time to assign some logic or instructions that will take a key press and translate it into a meaningful action, as well as check for overlaps or collisions between the player and the game level. See Listing 4.

There are basically three chunks of code there, a big one and two small ones. Let's tackle the big one first, which represents the process of taking keyboard input and converting it into an action for the player character. In this case, we are telling the player to move left if the left arrow is pressed, right if the right arrow is pressed, and to jump when the space bar is pressed. It looks a little complex, but that's because

flashpunk

If Flixel doesn't seem right for you, or if you find it frustrating, you may want to check out FlashPunk, which is also free. It was created by ChevyRay, a surprisingly young veteran of the Game Maker scene (see pg. 41 for more about Game Maker). It helps bring Flash development closer to Game Maker's style. Resources and more details about both are available at the Flash Game Dojo, which he and I co-created. Let's look at some of the major features of Flixel versus FlashPunk (note that this is not all of them).

FlashPunk
- :: Path-Based Movement
- :: Pathfinding Tools
- :: Intuitive Movement + Collision
- :: Style: Game Maker

Flixel
- :: Quad Trees
- :: Debug Console
- :: Automatic Tilemaps
- :: Style: C++/Java

```
override public function update():void
{
        player.acceleration.x = 0;
        if(FlxG.keys.LEFT)
                player.acceleration.x = -player.
                maxVelocity.x*4;
        if(FlxG.keys.RIGHT)
                player.acceleration.x = player.
                maxVelocity.x*4;
        if(FlxG.keys.SPACE && player.onFloor)
                player.velocity.y = -player.maxVelocity.y/2;

        super.update();

        FlxU.collide(level,player);
}
```

LISTING 4

we want that smooth Mario movement, so we're setting acceleration to 4 times the maxVelocity (yep, the same factor we used when we created the player). A lower acceleration means it would take longer for the player to reach his top speed, making the player feel heavy or clumsy. A high acceleration makes the character feel more twitchy and possibly overly responsive. This can be a tough thing to balance! Note, however, that the player only jumps if they are touching the floor, and that we set the velocity, not the acceleration.

```
super.update();
```

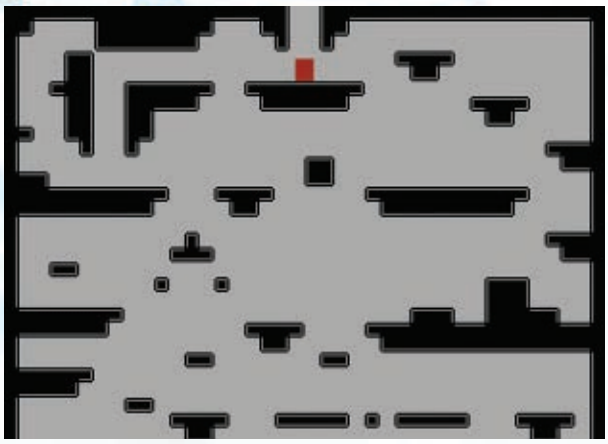
This line of code is tiny but very important. Like the FlxGame constructor, it is talking to the object you extended, in this case, FlxState. Calling `FlxState.update()` tells the game state to go through all the objects you added and call `update()` on each one. This is the call that actually moves them around the screen. `update()` checks for things like acceleration to be set, and then moves the object around for you.

```
FlxU.collide(level,player);
```

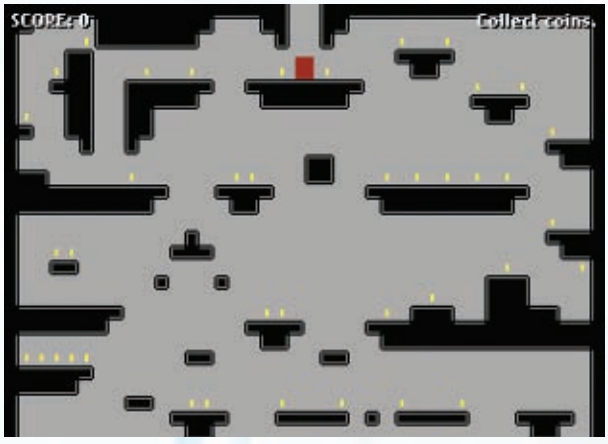
Finally, this critical line of code checks to see if the player overlaps the level anywhere, and if they do, pushes them out of the tile they're touching. This is what makes the level feel solid!

resources

EZPlatformer Demo	http://flixel.org/EZPlatformer
EZPlatformer Source Mirror	http://github.com/AdamAtomic/EZPlatformer
Flash Game Dojo	http://flashgamedojo.com
FlashDevelop	www.flashdevelop.org
Flash Builder Trial	www.adobe.com/go/try_flashbuilder
Flex SDK	http://opensource.adobe.com/wiki/display/flexsdk/Downloads
Senocular Tutorial	www.senocular.com/flash/tutorials/as3withmxmlc
Flixel	www.flixel.org
Flashpunk	www.flashpunk.net
TweenLite	www.greensock.com/tweenlite
Box2D	http://box2dflash.sourceforge.net



The same game environment, now with player and physical platforms.



The final game, complete with collectables, score, and an endgame requirement. Check out the source code online to get to this stage!

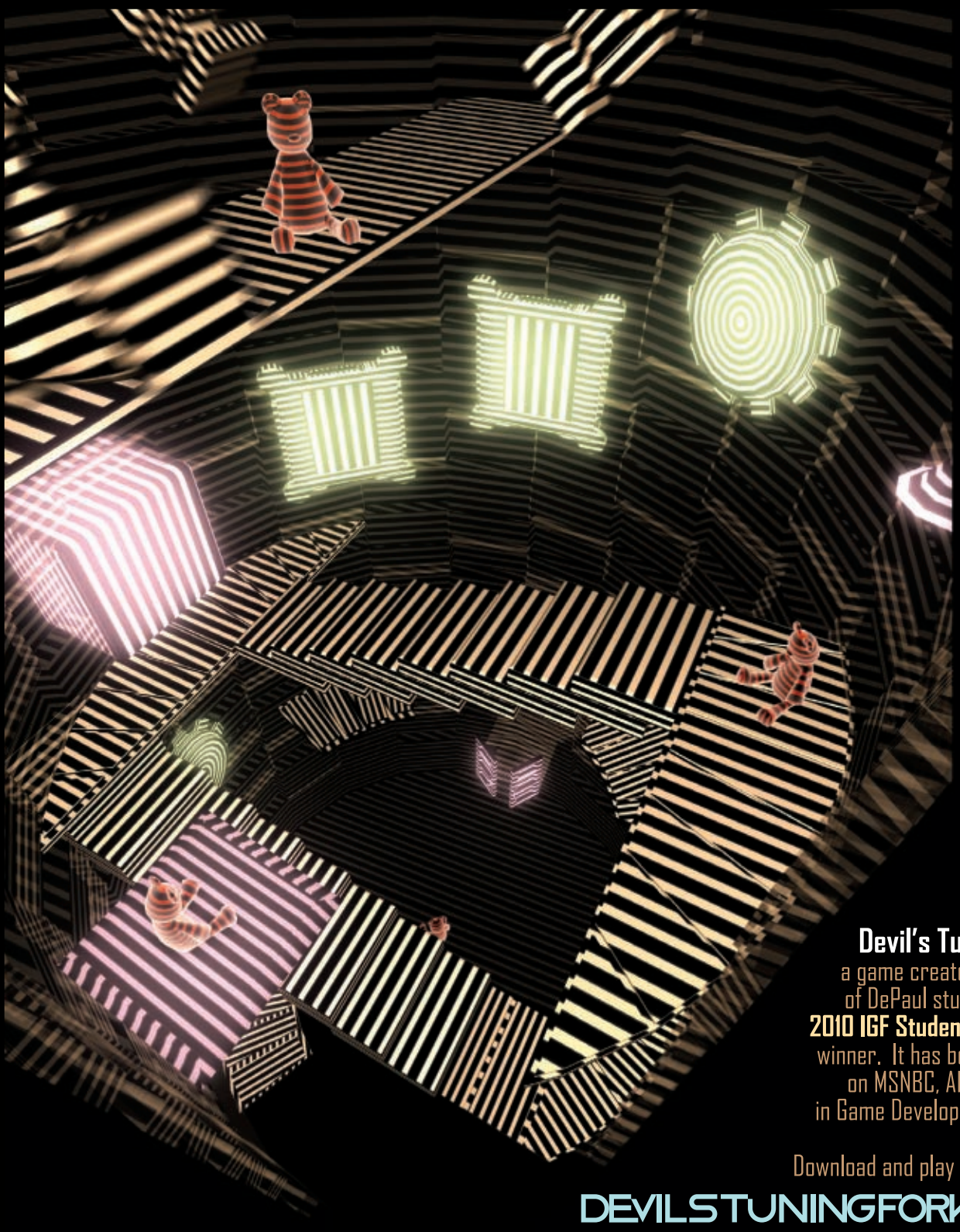
THAT'S IT!

» With some free tools and a few lines of code, it's really easy to get a simple game environment up and running in Flash. With just these two files we've created a cool, tilemap-based level for the player to explore, and created a player that you can control to do that exploring. That's all we have space for in this article, but all the source code for this project is online, and includes some important additions for real gameplay: some coins to collect, a win state, and a lose state. And if you want to learn more, or need more help, check out Flash Game Dojo, a joint project between myself and the creator of FlashPunk. It's a huge free resource for Flash game development!

HOMEWORK

» Try playing with the source code—change the level layout, change the speed at which the player moves. If you get bored with that, go online and pull down all the source code for the complete version of EZPlatformer and start fiddling with those variables. There is some extra code in that download, but if you read this far, you should have no problem figuring out what's going on with the new additions. Try adding a sound effect when you pick up coins, or adding an animated image to the player. Good luck!

ADAM SALTSMAN *is an independent game designer, co-founder of Semi Secret Software, and creator of Flixel. He makes games in Austin, TX with his wife Bekah and two idiot pug dogs.*



GAMEDEV.DEPAUL.EDU

Devil's Tuning Fork, a game created by a team of DePaul students, was a **2010 IGF Student Showcase** winner. It has been featured on MSNBC, ABC News and in Game Developer Magazine.

Download and play it for free at

DEVILSTUNINGFORK.COM

At DePaul you'll work with faculty who have extensive game industry experience, not to mention our faculty experts in animation, AI, computer graphics, cinema, game studies, screenwriting and sound design.

We offer graduate and undergraduate programs in **game programming**, **production**, **design**, and **animation**.

DEPAUL UNIVERSITY®

COLLEGE OF COMPUTING & DIGITAL MEDIA

Ign

IGNEOUS

BEN GABLE

WHAT IF YOU WERE ON A BRIDGE WHILE IT WAS EXPLODING? WHAT IF THIS BRIDGE WERE INSIDE AN erupting volcano? And what if one wrong step sent you to a horrible fiery doom? Well, you would have IGNEOUS, a junior/senior-level student project at DigiPen Institute of Technology in Redmond, WA.

It was during one of our many design meetings that we came up with this scenario. While we dismissed it as nothing more than a cool concept at first, it eventually came back around to inspire the core concept of our frantic action-platformer.

Team "Going Down In Flames" was born when four passionate students at DigiPen got together and decided that we wanted to "win IGF." IGF is the annual Independent Games Festival held every year at the Game Developers Conference. It's become a kind of tradition at the college these days. Every year, DigiPen students form into teams to create a game from scratch; it's a required class that all students must take (it is a game development college after all). Since the IGF has a student category, most teams set their sights on getting into the finals for a shot at winning big with the grand prize. We were one such team, and while we didn't have a solid game concept when we started out, one thing we did know was that we would do anything and everything we could to build a game worthy of the competition. We wanted to make something the student world had never seen before.

And so we started with a blank computer screen and slowly built IGNEOUS from scratch over the next 16 months. We laughed, we argued, and we ate at more buffets than most people do in a lifetime, but we never cried—or slept.

Ok that last one was a lie. But we never cried, ever.

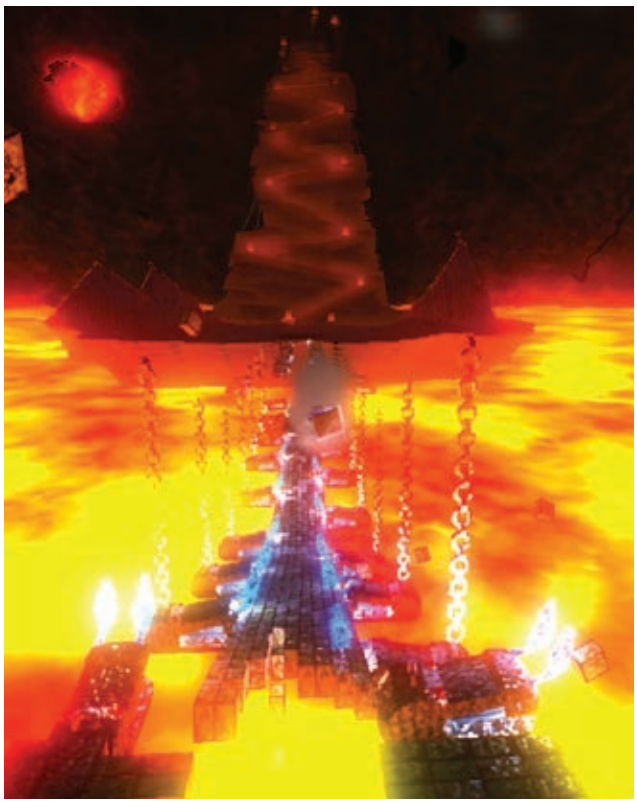
[WHAT WENT RIGHT]

1 | SIMPLICITY. Move and jump to get from point A to point B. That's all you do in IGNEOUS. There are no power-ups, special moves, or any other objectives that a player needs to worry about. IGNEOUS is a simple platformer at its core, and that's it. Since we were shooting for the IGF competition, we knew that judges were going to have hundreds of entries to get through and very little time with each one. We wanted them in the action immediately so that they "got" the game—no wasting time on tutorials or reading back stories.

We had experienced the pain of complexity with our initial design. Believe it or not, IGNEOUS started out as an arena-based combat game. Upon playing it for the first time, players would sit and read the fighting controls and then fumble around trying to remember what all of them were while hordes of lava blobs destroyed them. The learning curve was fairly steep and it took a while before players became comfortable enough to really get into the game. While some enjoyed this challenge and complexity, we knew it was going to hurt first impressions for competitions, and we just didn't like the idea of making a long and drawn-out tutorial.

It wasn't just the players and judges we were worried about—we were worried about ourselves as well. A combat game has many intricacies you have to get right if you want the game to be fun. You have to balance all the attacks, make player progression interesting, create lots of animations, and so forth. It was going to take a significant chunk of development time to make all this content, and even more to test and polish it. It was far too much for one producer and three developers to tackle while simultaneously taking upward of 15 college credits each.

So our motto for controls became "easy to pick up but hard to master," and our game design was simply "move and jump to get from point A to point B." It was a turning point for us. Instead of trying to force the game to be this complex beast of combat systems, we let the design naturally grow from our ultra minimalist concept. We tried to be as simple and straightforward as possible in every design aspect. After much iteration, we finally had something that our non-gaming parents could actually pick up and play. They didn't beat the game of course, but within 30 seconds of grabbing the controller, they were in the experience and knew exactly what they had to do and how to do it. For the team, this was the ultimate indication that we achieved that "easy to pick up" goal.



2 | BALANCING CHALLENGE. Our team had always struggled with this one. On one side, we had testers that would blow through the game without breaking a sweat. On the other, we had testers that couldn't make it past the first level without melting themselves in lava 50 times. It's truly eye-opening (and sometimes very frustrating) to see the incredible things that players can or cannot do. It can be anything from missing a turn that's seemingly very obvious, to naively falling into the same giant death pit over and over and over. With such a wide range of player skill out in the gaming world, we had to try and find a way to make IGNEOUS not too difficult, but not too easy, either. We didn't want potential judges bored or frustrated with the game.

Balancing the challenge turned out to be non-trivial, and after many days of tweaking, our answer to the problem was to have two modes of play: normal mode and impossible mode. Normal mode presented a reasonable difficulty curve and health system that gave players the ability to fall into lava or get torched by a laser and still keep rolling along. Of course, if they decided to go for a long swim, they would eventually die. However, their health would regenerate over time, hopefully letting them learn from their mistakes. Impossible mode, on the other hand, was exactly what it sounded like. Throwing out the difficulty curve entirely, if players touched any fire or lava, it resulted in instant death and a restart of the level. And we threw twice as much fire and lava at them as well. Impossible mode was not for the faint of heart.

While it took us a while to get to this answer to our difficulty problems, it turned out to work amazingly well for our game. Players that couldn't beat the game before now could, and the ones that thought it was too easy found themselves gritting their teeth in frustration. That's exactly what we wanted to see!

3 | WILLINGNESS TO THROW OUT EVERYTHING (EVEN THE ENTIRE GAME). As mentioned before, the team ended up completely redesigning the game about halfway through the project. Around the eight-month mark, we finally got to a point where we could playtest the actual game experience we were going for while we were still working on the arena-based combat design. This is when we found out it just wasn't as fun as we had imagined, and discovered it would be far too complex and time-consuming to go forward with that particular concept. So we simply threw it out and started back at square one. We still had our engine that we had been building, but as far as game concepts went, we had nothing. It took about a month and a half of constant brainstorming before we came back to the idea of running along an exploding bridge, and decided on the "point A to point B" design of the final game. While we were a little reluctant to just start over and keep nothing of our old combat design, we knew it wasn't fun, and it wouldn't get us into the IGF competition. It was a hard decision, but we made it and were far better off for doing so.

Even though a complete re-design of the game was by far our biggest change, the cutting didn't stop there. The final version of IGNEOUS has four complete levels. We actually made nine levels and ended up throwing out five of them, and when we say complete, we mean complete. Granted, they weren't as polished as the final four, but you could play through all of them from start to finish. A few of these were different versions of the current

BLIZZARD IS HIRING

We are actively recruiting across all disciplines for the following locations:

IRVINE, CALIFORNIA | AUSTIN, TEXAS | VELIZY, FRANCE | CORK, IRELAND
SINGAPORE | SHANGHAI, CHINA | TAIPEI, TAIWAN | SEOUL, SOUTH KOREA
SAO PAULO, BRAZIL | BUENOS AIRES, ARGENTINA | MEXICO CITY, MEXICO

jobs.blizzard.com

levels and a few were completely new scenarios. The most significant one was a level where the player rides down a raging lava river on giant rocks and pillars. Our physics developer actually incorporated buoyancy into his custom physics engine just to make it possible to build this level. While it sounded really cool on paper, we found there were several problems with it that detracted from the fun factor once we got it up and running. The floating rocks didn't quite behave how we wanted them to, it was extremely difficult, and navigating from rock to rock on the lava river was clunky. So again, we ended up throwing it out completely.

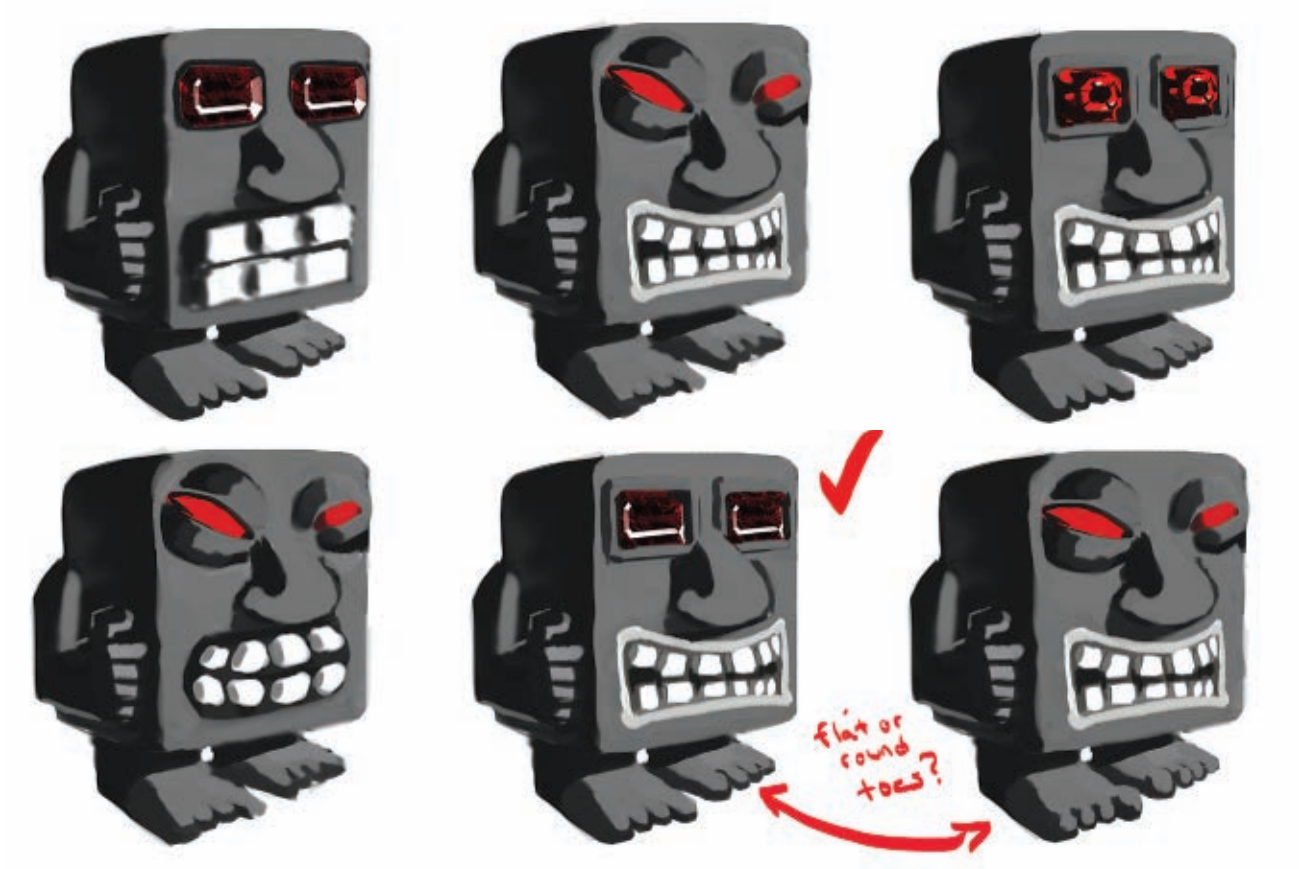
As a student team, we had a little more flexibility when it came to throwing things out. We didn't have a publisher and a multimillion dollar contract to adhere to, but it still meant a lot of time lost building content that wouldn't be in the final version. However, the team felt that keeping the truly great chunks of gameplay and throwing out the simply good ones helped immensely in delivering a better experience overall, which is something players responded to during our many playtest sessions.

4 | PLAYTESTING. From the early stages, the team knew that if we wanted to get in to IGF, our game had to be exceptionally fun. We wanted people to love our game and want more of it. If we could get it to that point, then we'd have a chance at making it into the competition. In pre-production, we committed ourselves to getting a playable version ready as soon as possible. We were going to need feedback on our design so we could adjust and change accordingly. At DigiPen, most student teams wait until the second semester of development to start playtesting. For us, this was far too late in the cycle.

It was a push to get things up and running together within a couple of months, but it really paid off for us. We got a simple prototype of our first combat design built from scratch in a couple of months and started playtesting every Friday in the back of one of DigiPen's large computer labs. We used Google Docs to record player feedback and observe what players were saying about the game. After each session, we'd look at input, discuss the changes we'd make, and incorporate them into the design. Our goal was to never test the same build twice and always have a new gameplay element in for people to play around with. While it might sound like an easy thing to do, it was quite difficult at times, especially when we had three programming assignments and a math quiz due the following week! And another two the week after. And then five midterms after that.

Even though the process was a little hectic, it was only through playtesting that we discovered the flaws of our first design and decided to throw it out. On our second design, we got invaluable feedback on our level layouts and difficulty curves. It really allowed us to refine each level to a very smooth and polished experience that eased players into the game and then gave them a true challenge.

5 | NOT WORKING ON THE GAME. Our development schedule was pretty regular. Monday through Thursday, the team would get to the lab around 9–10 AM and leave at midnight. Friday was our playtesting and homework day for those pesky "other" classes. Saturday and Sunday were a little more relaxed, with the team coming in and working from noon to 8 PM. IGNEOUS consumed our every waking moment. However, it became very clear early on that we couldn't do this week after week without blowing off some steam. So Friday nights became our fun nights. Barbecues, movies, video games, card games, restaurants—the IGNEOUS team and sister team Kabloom would play it up with



generous helpings of food and drink. Everything we did was classified as a "team building exercise," and it really worked in relieving tension among the developers and building a sense of camaraderie.

As time went on, we developed other regular exercises. One of them was the formation of the Buffet Strike Force, which aimed to visit a new buffet every other week. Another was our Washington Hikers group,where we would climb various mountains in and around Washington state's Central Cascades. We knew that if we wanted to make IGNEOUS fun, we had to have fun doing it. And the team feels that our work hard, play harder mentality paid off in spades.

[WHAT WENT WRONG]

1 | DISREGARDING THE NEED FOR ART. Although DigiPen has both a game programming and game art degree, the two have been very separate until recent times. In fact, for the past four years the two groups studied at different campuses because one could not hold both programs at the same time. Because of this, most of the student game teams relied on "programmer art" for their projects since the artists did not take the same game development class that we did. The option to recruit artists did exist, but this was a path that the IGNEOUS team originally disregarded.

Our team wanted to remain small and agile and the idea of relying on art from people across town screamed "bottleneck" to us. So we decided to not look for artists and focus on developing the game with art assets that we created. If we needed art later, we would simply recruit them then. Since everything in our game was physically simulated and our main character was a stone cube, we didn't need animation, because we could just let the physics engine do all the work. And since IGNEOUS takes place inside of a volcano, the plan was to use lots of particle systems and post-processing effects to create realistic fire, lava, and explosions.

For the most part, we were very happy with how our game looked despite not having artists. However, as development continued, student teams that had recruited artists started to look better and better. We soon found ourselves at a point where we had these amazing environments we wanted to create, such as a Mayan city inside of the volcano, but we couldn't do a lot of it because we didn't have the skill to create good looking in-game models.

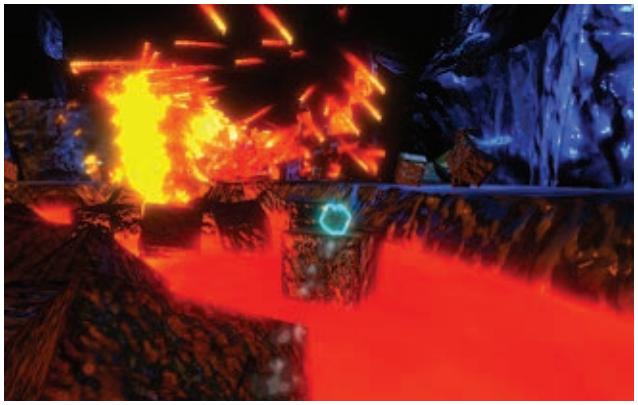
We decided it would be a good time to start recruiting, but soon found out that most artists were already taken by other teams or swamped with other classes. Since we hadn't found someone willing to dedicate to the project up front, any possible leads were mostly unreliable.

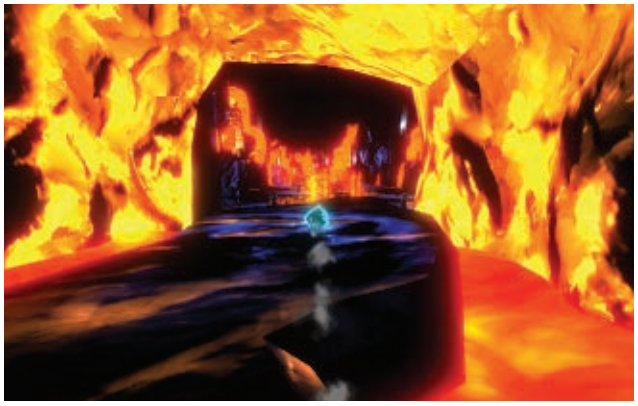
The team eventually did get two artists to sign onto the project during the final three months of development, but the process was rocky. Communication was hindered by the fact that none of the programmers knew how long or what it took to actually make a real art asset. Without a solid content pipeline between the artists and programmers, we could only get a few main assets created in time for submission into the IGF. One example was the player tiki totem.

While the IGNEOUS team still prides itself on how the final game looks, and the artists we did recruit did amazing work, the potential for it to have looked much better was lost due to the low importance we placed on the entire process at the start of development.

2 | FEATURE CREEP. Making major changes or adding large amounts of content at the end of the development cycle was both a strength and a weakness for the IGNEOUS team. Since we had always been willing to throw out things that weren't working in the game, it meant there was a "hole" that needed to be filled or cut whenever we did so. Most of the time, we tried to fill it with something entirely new, and ended up throwing that out as well and just cutting back the scale of the game. There were two particular scenarios where changing or adding something at the last moment came back to bite us.

CONTINUED ON PAGE 38





CONTINUED FROM PAGE 37

The first and most embarrassing instance was during a special presentation. Legendary FINAL FANTASY composer Nobuo Uematsu visited DigiPen while he was in Seattle for a FINAL FANTASY concert. Since this was a special guest, the top DigiPen teams were asked to present their games to him and IGNEOUS was one of them. The technical director had just switched the audio engine to use FMOD Designer files for all of our sound effects and music. Since we were presenting to a composer, we decided to update a few of the really bad sound effects and add newer versions of the music tracks. While presenting, the game started to freeze up and eventually crashed—our first ever crash during a presentation. Later on, the problem was traced back to an incompatible compression format for one of the changed sound effects. Very embarrassing and ironic.

The second scenario was much more severe. Our two difficulty modes weren't actually added until the final weeks before the game was due for the IGF competition. Adding them resulted in a bug in our menu system that caused changing options to not work correctly. The final week before submission, we fixed the bug and submitted our final build. Everything worked fine, right? As it turned out, fixing that menu bug caused a much more severe bug that we failed to catch before sending out the final game. Changing *any* option in the menu system would automatically up the difficulty to impossible mode. We knew that most judges would never get past the first level of impossible mode on their first time playing. It was a huge screw-up on our part that would destroy their experience of the game. Luckily, the problem boiled down to one number in one line of code that needed to be changed. We were able to get a fixed version off to the judges, but still hit ourselves for potentially ruining our chances of getting into the competition with such a stupid mistake.

3 | CRUNCHING UNTIL NEAR DEATH. Needless to say, we worked on the game a lot. Every milestone turned into a major crunch that encompassed all the bugs that needed to be fixed and the design changes that had been put off. These crunches lasted anywhere from 24 hours to 3 days, during which time the team would sit around our dining room table, working until the list of things we came up with before crunch was entirely crossed off. Depending on how long we had been going, we'd take 2–3 hour naps every now and then and when we got hungry, we'd run to the local store for frozen pizzas or hit up Jack in the Box. When team IGNEOUS crunched, we *crunched.* It usually ended with us being totally out of commission for the following couple of days as well.

While we got a lot of stuff done during these sessions, they were grueling. We never crunched on trivial stuff, either. Everything on our list had to be done or we'd only be hurting ourselves in the long run. We realize that most of these crunches could have easily been mitigated by spreading out the work over the entire cycle rather than the week before a milestone was due. A lot of the issues came from our relative inexperience when it came to making a game such as IGNEOUS. It was by far the biggest and most involved project any of the team had ever worked on. Knowing what the team knows now, we'd be much more committed to fixing bugs and implementing design changes right away rather than throwing them on a to-do list.

4 | LACK OF DESIGN FORESIGHT. When we set out on our path to create an IGF-worthy title, we were naïve when it came to a game design. We went through numerous ideas up front that involved things like balancing platforms and weightless physics puzzles, but they were all quickly thrown out. We didn't come up with an idea that we really liked until we found our first concept of an arena-based combat game set inside a volcano. When we all liked the sound of it and decided to prototype it out, we never gave the design any more thought—we became consumed with our goal of prototyping a playable demo. The idea never got past the concept stage. We never stopped to think about how complex the combat would be or how difficult it would be for players to pick up and play. It was only after we had built the entire prototype and started playtesting that we started to think about these things and see some of the problems firsthand when players gave us their feedback. It was almost halfway through the 16-month development cycle when we ended up making the decision to throw out that design.

While we did spend most of that time building game systems that we would eventually need anyway, there were some that went entirely unused. The team still thinks about how awesome our final design would have been if we had set out with this idea in the first place. Granted, much of what we experienced in building the first design played a role in us coming up with the final one, but we all agree that we could have foreseen the problems associated with our combat concept much earlier on had we not been so eager to prototype it right away.

5 | TOO SHORT. This was an interesting problem for the IGNEOUS team. The final game is a mere four levels long, and the first level is essentially a tutorial. So in the end, players have three levels of action to conquer before the credits roll. To many, this was far too little.

By far the greatest complaint we've received from emails, comments, and forum posts is that IGNEOUS was too short. Where this becomes interesting is that we knew the game was on the short side and had hoped players would walk away wanting more from the title. It could only be a good thing when people want to play more of your design, right? At least, that was part of our reasoning for impressing judges for the IGF competition.

While it's a very good problem to have, there were many levels and ideas that the team really wanted to get in but ultimately cut due to time constraints. As mentioned before, we created nine fully playable levels but only polished four for the final version. Beyond that, there were at least three other concepts for levels that the team played around with, but didn't even attempt because of the sheer scope and scale, including a level that took place outside of the volcano! As much as we may have tried, we learned fast that we were simply a student team incapable of building things on a triple-A scale, especially with the time frame we had to do it in.

[READY TO ERUPT]

» In the end, IGNEOUS was an amazing success for our team. While we didn't quite achieve our "win IGF" goal, we made it to the student finalists and had a blast participating in the competition. The viral response from players across the Internet, at PAX, and at GDC was extremely positive. By far the greatest reward for us was seeing so many people enjoy IGNEOUS. Looking back at how we started out and where the project ended, we've learned magnitudes about what it truly takes to make a fun and polished game. The team has split for now and there are no current plans for another iteration of IGNEOUS, but who knows what the future may hold. In our eyes, this was only the beginning!

BEN GABLE *was the producer for IGNEOUS. Check out www.igneousgame.com to download the game!*

GAME DATA

IGNEOUS

PUBLISHER DigiPen Institute of Technology

DEVELOPER Going Down In Flames

NUMBER OF DEVELOPERS 1 Producer, 3 programmers, 2 artists

BUDGET $0.00

LENGTH OF DEVELOPMENT 16 Months

RELEASE DATE November 1, 2009

TOOLS Visual Studio 05/08, Tortoise SVN, 3DS MAX, GIMP, FL Studio, FMOD Designer

LANGUAGES C/C++, Lua

MIDDLEWARE FMOD Ex

PLATFORM PC

Game Degrees for

Creative Minds℠

Want to unleash your creative powers through a career in the gaming industry? Full Sail University's game degree programs can show you the tools that today's studios use to bring their projects to life. If you're ready for a career path in games, start your journey today – and learn why Full Sail is considered a leader in game education.

- **Game Design**
  Master's Degree
  Campus
- **Game Art**
  Bachelor's Degree
  Campus & Online
- **Game Development**
  Bachelor's Degree
  Campus
- **Game Design**
  Bachelor's Degree
  Online

THE HARVARD OF GAME SCHOOLS

— Tips & Tricks Magazine

ONE OF THE TOP FIVE GAME DEGREE PROGRAMS IN THE WORLD

— Electronic Gaming Monthly

FULL SAIL UNIVERSITY®

fullsail.edu

800.226.7625 • 3300 University Boulevard • Winter Park, FL 32792

Financial aid available to those who qualify • Career development assistance • Accredited University, ACCSC

MAKE GAMES NOT WAR

A SHOOTING GAME TUTORIAL FOR GAME MAKER

JESSE VENBRUX

Game Maker is a great tool for beginners. It's where I got my start, and where I "grew up" in terms of game design, helped by the Game Maker community. I'm perhaps best known for the game KAROSHI, and am well established within the Game Maker community. It's for that reason that I've been asked to create this tutorial for making a basic shooting game with the software. I am not a programmer at heart. All I want is to entertain the player and create fresh experiences. I believe Game Maker is the program that has allowed me to do that best, while also helping me to develop my skills. I hope others will also recognize the power a game development tool can possess.

WHAT IS GAME MAKER?

» Game Maker is a cheap computer program for game development with an intuitive interface (the pro version costs $25). It was originally developed for kids to give them a sense of the basics of programming, but has since evolved to become a darling of the indie games community. It's an especially good tool for 2D game development and rapid prototyping in general.

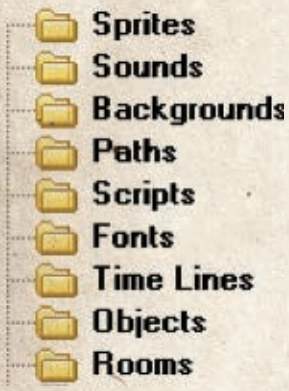
I particularly recommend Game Maker if you want to learn more about game design. Game Maker is the fastest and easiest way to get something running. The more games you make, the more you'll learn—after all, game design is best learned by actually designing games. It's also not limited to any specific game type or genre, allowing full control over what you wish to make. >>>

If you want to learn to code, it's also a great place to start getting the hang of programming logic. If you're planning to pursue a career in game programming though, you're better off learning C++.

If you want to get an idea of what's possible with the software, check out the video "100 Game Maker games in 10 minutes." Another good example is Cactus's TUNING, which won this year's Nuovo Award at the Independent Games Festival. See Resources for links to both.

STRUCTURE OF GAME MAKER

» Game Maker's main interface consists of a few folders you can see below.



Let's break down those folders.

Sprites. This is where we store the images that are used for our game (except for backgrounds). Sprites can be single images or animations (several images).

Sounds. Here, we can put sound effects as well as music tracks for your game.

Backgrounds. Backgrounds are usually larger images that are used as ... well, backgrounds, in levels. In this folder, we can also put tilesets.

Paths. These are paths that objects in our game can follow to make them move. Paths can be drawn in the path editor and can be any shape you want to make them. Useful for creating enemy move patterns and the like.

Scripts. Scripts are separate pieces of code you can use across all your objects. In programming terms, they are what you'd call "functions."

Fonts. Fonts are fonts! These are type settings for the text in your game.

Time Lines. Time lines can be used to execute code at certain times / time intervals.

Objects. This is an important one, and you'll be using it the most. Objects are the things that your game consists of. You could that say your game consists simply of sprites, but sprites are nothing without code attached to make them do stuff. This is where objects come in: they each have an assigned sprite and hold all the code.

Rooms. Rooms are the levels, menus, and so on (screens) for your game. The room editor in Game Maker is especially useful because it can also function as a level editor.

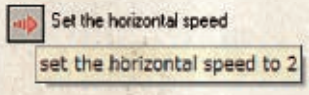
GAME MAKER LANGUAGE

» Since Game Maker is a great place to start for those with no programming experience (like me when I started), I'll be showing each step using the drag and drop icons. In addition, I'll show how the same steps can be done in Game Maker's own programming language (GML).

There are a number of "drag & drop icons" you can use; they are divided into separate tabs (left).

That shows just one tab of the available icons. Although you can do a lot with these, in the end, you can still do more by using regular code. However, for beginners, it's enough to make simple games.

Let's get into this "programming" thing. Here's a simple example: The icon below represents setting the horizontal speed of an object to a certain value.



This makes an object move horizontally across the screen at a speed of 2 pixels per "step," where 1 step is 1 frame. Games usually run at 30 or 60 frames per second.

In code, it looks like this:

```
hspeed = 2
```

GML is relatively simple compared to other languages, and doesn't need as many parentheses or punctuation as others do. However, if you're familiar with programming and like to do it the "real" way, you can:

```
hspeed = 2;
```

Also, the following code examples both work fine:

```
if variable = true hspeed = 2

if (variable==true) {
 hspeed = 2;
}
```

Just write it the way you like it. In this tutorial, I'll be using the simplest form. Also note that you don't need to specify any data types in Game Maker.

EVENTS AND ACTIONS

» An event is something that happens in the game (e.g. two instances collide, or a key is pressed). Some events can trigger a certain behavior. This behavior is defined in actions. The drag & drop icons, as well as the code you just saw, were all actions.

Whereas actions determine *what* happens, events determine *when* it happens. In the above image you can see some of Game Maker's events. As examples, you can let something happen when a key is pressed, when an instance is destroyed, when two instances collide, or every frame (called "Step" here). Actions are always assigned to events, not the other way around. So first you decide when it should happen, then you decide what.

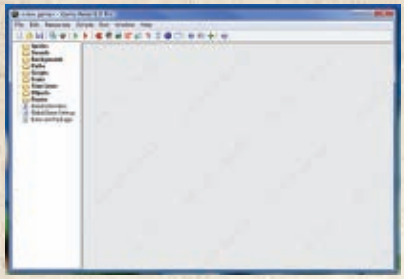
THE TUTORIAL

In this tutorial, we'll be making a basic shooting game. It teaches you the basics of Game Maker in the simplest way. If you're relatively new to programming, don't worry; just follow these 16 steps. If you have experience programming, it'll be a breeze (and probably too easy, but that's better than too hard). It shouldn't take more than 15 minutes. What you need to get started is a PC with Windows (a Mac version is coming—by the time you read this, it may even be out), the Game Maker program, and some images which I've provided for the tutorial. After this tutorial, you'll have an understanding of the basics of Game Maker. From there, you'll be able to start making more complicated games.

1 First things first, download and install Game Maker. You can download the lite version from YoYo Games (see Resources—you don't need the pro version for this game).

If you like Game Maker, I recommend going Pro; it's only $25 and definitely worth it. Run the installer you've just downloaded, and after it's finished, start Game Maker. You may be asked

whether you want to use "advanced mode" or "simple mode." Go with "advanced," because it gives you a few more folder options. Your view should look something like what you see below.

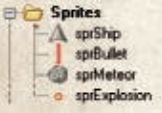


You might also see some windows open, like "News," which we don't need now, so you can close them.

2 Next, we need some images so we can actually display something on-screen. In other game creation programs, you might have to set up your project first or write the basics for your engine. Not in Game Maker, we can start right away. We'll need the following images:

- spaceship
- bullet
- meteor
- explosion (animated)

You can make your own images with GM's built-in sprite editor, or use any other program you prefer. To make it easy for you though, you can download a simple sprite pack I've prepared from the *Game Developer* web site. These images also come with the program, as do many others. After you've downloaded the images resource file (or created your own), choose "Import Resources" from the File menu, and select the downloaded file. You'll see that the sprites will be added to your sprites folder (left).



If you've made your own sprites, you'll have to add or create each manually. Sprites can be made with any graphics editing program as long as you can export them as image files. PNG, GIF and BMP all work. I advise to use the PNG format since it can hold transparency. You can also make sprites with Game Maker's built in sprite editor by clicking the "Edit" button in the sprite window, but it's not as good as professional image editing software. You can add a sprite by either clicking the red Pac-Man in the top bar or right-clicking on the Sprites folder and selecting "Create Sprite."

3 Now we're going to make our first object, so the images we have can actually start doing something on-screen. Click on the blue ball icon in the top bar, or right-click on the Objects folder and select "Create Object." You should see the following window pop up:

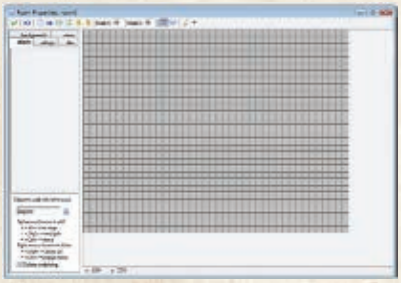
Do you see the "sprite" part in the top left? Click on or next to where it says "<no sprite>" to assign a sprite to the object. Choose the spaceship one, because this is going to be our spaceship! Although it's unnecessary for this tutorial, it's usually helpful to give the object a name, like "objSpaceship" or anything else you fancy. That's it! We now have a spaceship!

Press OK.

4 Run the game. You can do so by pressing the green arrow icon in the top bar, or F5 on your keyboard. If all is well, you'll see the below error. Don't worry, this happens to me all the time. Basically, you can't have a game without there being some screen for it to exist on. This means we need to add a Room.

5 Click the room icon (it looks like a small window) in the top bar, or right-click on the Rooms folder and select "Create Room."

The below window will show up.



That grey field is the actual level. Now you can try to run the game again. It should run, but you won't see the ship yet. To fix this, first click somewhere in the empty field in the left side of the room editor (under "Objects") and select the spaceship object (it should be selected by default). Then, click anywhere in the room to add it to our game.

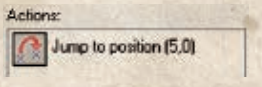
6 Now run the game. You'll see it working with our spaceship. It's not doing much, however. By the way, if you're having problems, you can download a correct version of the game at every step, and compare to see what you missed. See Resources.

7 Time to start coding! If you have the object window still open (if not simply double-click the spaceship object), you'll notice that there are two big empty fields. One for events, and one for actions. Now click on the "Add Event" button. You'll see the window displayed below.

Select the event called "Keyboard," and choose "<right>." This is the event for whenever the Right Arrow Key is being held down. Now we're going to add some actions in the actions field. Select the one called "Jump to Position" (right).

Simply drag it onto the actions field. Now, a window will pop up which we can use to set where we want our spaceship to move (see below).

For x, fill in 5. This means our spaceship will move 5 pixels over the x-axis, which is to the right (corresponding with the Right Arrow Key). Keep the y value at 0, we're not moving up or down. Click OK. When you're done, you should see the following in your object window (below).



CONTINUED ON PAGE 45

Careers@Singapore: Interactive & Digital Media

Create original content for a global market

Singapore aims to become an interactive & digital media (IDM) capital which generates original content for the global market. Major and homegrown game developers such as Electronic Arts, LucasArts, Ubisoft, Tecmo Koei, IGG, Rainbow S.p.A, Singapore-MIT GAMBIT Game Lab (GAMBIT), Softworld (Zealot), Mikoishi, Matchmove Games and Ratloop Asia are in Singapore, alongside game service providers and payment service providers.

These are part of our strategies to position ourselves as the regional games hub for the development and distribution of online and mobile games harnessing on emerging technology.

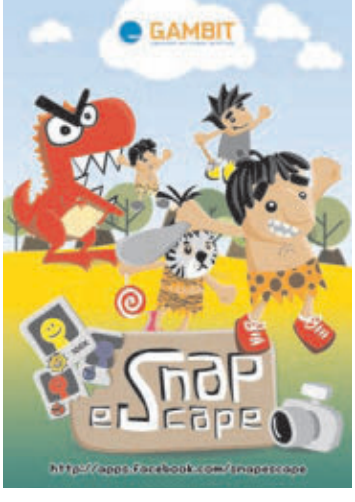
Some of the exciting opportunities for talent in Singapore include:

GAMBIT

- Assistant Producer
- Game Artist
- Programmer
- QA Lead
- Research Executive

UBISOFT

- Producer
- Senior Game Designer
- Senior Network Engineer (Online Experienced Tech Lead)



"The quality of life is just amazing. Everything is super-efficient — especially the public transport system which is very organised and very cheap, the food is incredible, and there's just so much energy in the air… Everybody speaks English."

Technical Director Claude Langlais, transferred from Ubisoft Montreal to Ubisoft Singapore two years ago.

"It not only opened my eyes to the academic, research, and development of digital media in Singapore, but I had fun learning about the city, its people, its food, and its culture. I gained so much from the trip — the knowledge, the experiences and all my new friends"

Syracuse University student Jasy Liew, who was in Singapore and introduced to its IDM industry and life in the city-state.

About Singapore

The Singapore government has identified the Interactive & Digital Media (IDM) industry as a key growth industry. It is targeted to generate 10,000 new jobs and create S$10 billion worth of value added by 2015. The game industry is one of the key IDM sectors that we are seeking to grow, as we work towards our aim of becoming an Asian IDM capital from which original content is created for global consumption.

Visit **www.jobsatsingapore.sg** to apply for the latest positions

Go to **www.contactsingapore.sg** for information on working and living in Singapore

THE DOT IS HOT

Contact Singapore
contactsingapore.sg

CONTINUED FROM PAGE 43

This is the equivalent in code:

```
x = 5
y = 0
```

Here's a side note for those who are going to be coding: To be able to write code, you first need to drag the code icon onto the actions field. You can find it under the "Control" tab (tabs are at the very right of the object window). See the icon to the left.

After dragging it onto the actions field it opens a code window automatically.

Okay, back to these actions. What exactly are we doing here? So, when the Right Arrow Key is pressed ... the object jumps to the position (5,0), correct? That position is somewhere on the top left of your window, since that's (0,0). (The bottom right would be (640,480)). Therefore, in the Jump To Position window, make sure to tick the "Relative" checkbox. This will make the object jump to a position *relative* to its current one. In other words, were the position (100,100), it would next jump to (105,100) instead of (5,0). In code:

```
x += 5
y += 0
```

Note: For those that don't know, "x += 5" is a shorter notation for "x = x + 5".

Next up, try to repeat the previous bits, but this time for the Left Arrow Key. For x, you should fill in -5.

8 If you run the game now, it should work correctly. Next, we're going to make our spaceship shoot some bullets. Add a new object and assign it the bullet sprite. Then click "Add Event" and choose the "Create" event.

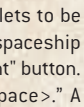
This event is executed only at the moment the object is added to the game. For example, if this were the spaceship object, it would be executed right when you start the game. Add the icon you see to the left to the actions field.

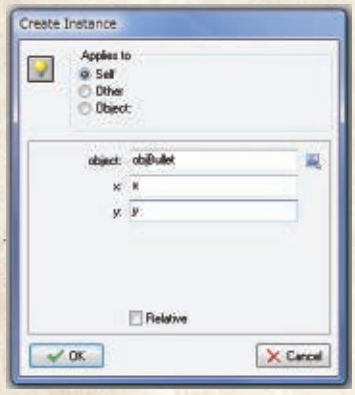
For the vertical speed (vspeed) value, fill in something like -10. Vertically, everything goes from top to bottom in Game Maker, so a negative speed moves upward. This action gives the bullet a vertical speed. We only need to set it once, so the Create event is the logical option here.

Equivalent in code:

```
vspeed = -10
```

9 We've made a bullet, but it's not in the game yet. We're not going to add this one directly in the room editor, like we did with the spaceship, because we want bullets to be created in a different way. Open the spaceship object window, and click the "Add Event" button. Select "Key Press" and choose "<Space>." A Key Press event is only executed when you start pressing a certain key, not when the key is held down (like the general "Keyboard" event). So when the Space bar is pressed, a bullet will be released. Drag the "Create Instance" icon onto the actions field. You can find it under the "main1" tab. A new window will show up (below).





For "Object:" select the bullet object. For x and y, simply fill in x and y (just like above). This means that the bullet will be created at the x and y position *of the ship*. (We're coding within the ship object, so the x and y here refer to the spaceship itself.)

Equivalent in code:

```
instance_create( x, y, objBullet)
```

10 Run the game. If everything went correctly you should be able to shoot bullets now! But what's the point of shooting bullets if there's nothing to shoot at? Add another object. This is going to be a meteorite. Make sure to assign it the meteorite sprite, or else we won't see them (but that could be a fun game too?). Meteorites will come from the top of the screen and move down. I'll leave this little part to you. You just need to copy what you did for the bullet, but give it a positive vertical speed.

11 Done? Good. Next, we need to add a way to make meteorites appear in the game. This is a little more complicated. You'll need to add another object. This object is going to create the meteorites for us. To do this, we are going to use the Alarm event. This is simply a timer you can set to a number of steps, after which it goes off and executes the code that's assigned to it.

Steps are basically frames. Default Game Maker games run at 30 frames per second, so 30 steps would be one second. But that's the default, of course. Don't worry, Game Maker can easily handle 60 fps. Add a Create event to the new object. Now add the "Set Alarm" action to the actions field (right). This action can be found under the "main2" tab. Set the alarm to 30 steps.

Equivalent in code:

```
alarm[0] = 30
```

After this object is created, it will take 30 steps (1 second) for the Alarm to go off. However we haven't made the alarm itself yet, so let's do that. Click the "Add Event" button and pick the Alarm event. You can choose from up to 12, but just pick "Alarm 0." Now, when the alarm goes off, we'll make a meteorite appear on screen. Just like the bullet. It's the Create icon again (right).

For x, fill in "random(640)" and leave y at 0. What this does is create a meteorite at a random x position between 0 and 640. 640 is Game Maker's default room width. Don't forget to assign the meteorite object as the object to create!

Equivalent in code:

```
instance_create( random(640), 0,
objMeteorite)
```

Next up, you'll have to add the same thing we put in the Create event to the Alarm event as well. We have to reset the alarm every time it goes off, or else it will only create a meteorite once. And finally, the object has to be added to the game. We only need it once and we need it from the

gamemaker pros and cons

PROS

1) The drag and drop interface is intuitive and easy to understand for beginners.
2) Many helpful functions for game creation.
3) Get your game up and running in no time.
4) The room editor is great for quick level design.
5) No genre restrictions; you can make virtually any (2D) game.
6) Great community and support.

CONS

1) Game Maker games can't run natively in a browser (like Flash). There is a plug-in, but players will have to install it first.
2) Game Maker games only work on Windows (a Mac version is in the works).
3) 3D functions are limited.

start, so put it somewhere on the stage using the room editor.

12 Run the game and see if it does what you expect. Now we almost have our game! But no matter how many bullets you shoot, you probably won't destroy many meteorites. What we'll cover next are Collision events. Open the meteorite object window and add a Collision event. Choose the bullet object for it to collide with (below).

For actions, simply use the "Destroy the Instance" icon (it's on the "main1" tab, also seen left).

Equivalent in code:

```
instance_destroy()
```

13 If you run the game now, you should be able to destroy meteorites. Now we need to add some explosions. Create a new object for the explosion and assign it the explosion sprite. Make sure you create it when the enemy object is destroyed (that would be the collision event we just made). I hope you know how to do this by now. Tip: Create it at the x and y position of the meteorite. If we're going too fast, remember you can always download the current state of the game from the link in Resources.

14 The explosion sprite is not a single image, but a set. It's animated. The object will automatically play through the sub-images after creation. If we leave it like this, explosions will be created but never destroyed; they'll loop repeatedly. To fix this, we use a special event called "Animation End." You can find it under "Other" in the event window. Assign it the "Destroy the Instance" action.

15 We're going to add a score. Within the collision event between the meteorite and the bullet, add the "set a variable" icon (on the "control" tab, see the icon on the right).

We're going to increase the score when you destroy a meteorite. For variable, fill in "score," and for value, fill in something like "100." Make sure to tick the "Relative" box, so that the score adds up (as below). Otherwise, it will just reset to 100 each time the action is executed.

Equivalent in code:

```
score += 100
```

16 Run the game. Everything working fine? We're almost done. All the game is missing now is an end condition. To add this in the simplest way, we'll just end the game once our spaceship hits a meteorite.

Add a Collision event in the ship object and choose the meteorite object to collide with. We're going to add the following 2 actions (right). The first will automatically display a high score listing and the second simply restarts the game.

Equivalent in code:

```
highscore_show(score) game_restart()
```

And with that, you're done! If you're interested in trying more, and want to mess with somewhat more complex tutorials, check out some of the free ones on the Game Maker site. They should be easy now that you have some basic familiarity.

JESSE VENBRUX *is an active member of the Game Maker community, and is best know for his game* KAROSHI. *You can see more of his work at www.venbrux.com.*

resources

100 GAME MAKER GAMES IN 10 MINUTES
www.youtube.com/watch?v=Nek-gMLunj8

TUNING www.youtube.com/watch?v=ji-0PheJGdk

GAME MAKER SITE www.yoyogames.com/make

GAME DEVELOPER CODE SECTION, FOR SPRITE PACK AND CORRECT STAGES OF THE PROJECT
www.gdmag.com/resources/code.htm

GAME MAKER TUTORIALS
www.yoyogames.com/make/tutorials

10 stude

Student games are taking an increasingly prominent position in the game industry. As students use their time in school to create interesting and meaningful projects, the game industry at large is taking notice, often bringing student works to the commercial sphere. Here, we present 10 student games from the last few years that had an important impact on the game world. *— Tim W. Boon*

CLOSURE

Developers: Tyler Glaiel and Jon Schubbe
Web site: www.closuregame.com
Institute: DigiPen Institute of Technology

Tyler Glaiel and Jon Schubbe's puzzle platformer was a finalist in three IGF main competition categories this year (technical excellence, the Nuovo Award and excellence in audio), and is still being developed by Glaiel while he is studying at DigiPen. Though the game is not a thesis project for his course, the institution was happy to acknowledge Glaiel's achievements in the competition.

While it might be some time before the competition version is released by the duo, an early Flash prototype posted at the end of January 2009 is still available to play online.

CONTINUITY

Developer: Ragtime Games
Web site: www.continuitygame.com/about.html
Institute: Chalmers University of Technology, Gothenburg, Sweden

Not only was CONTINUITY chosen as one of the ten IGF student showcase finalists in 2010 (out of a total of 190 entries), this clever puzzle platformer even won the award for the year. The original Flash prototype was developed over a period of 16 weeks by a team of four students, under the supervision of a university professor at the Chalmers University of Technology in Gothenburg, Sweden.

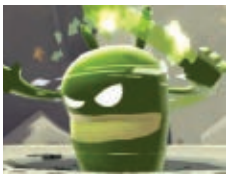
Ragtime Games did state that it is working on an iPhone version of CONTINUITY, although no release date has been set as the team has yet to graduate from the Interactive Design masters program.

CRAYON PHYSICS DELUXE

Developer: Petri Purho
Web site: www.crayonphysics.com
Institute : Helsinki Polytechnic, Finland

Unlike other entries on this list, CRAYON PHYSICS DELUXE actually derailed Petri Purho's education at Helsinki Polytechnic, where he had been studying computer science for some time. The story was that he took half a year's break from school to update the first release of CRAYON PHYSICS after receiving lots of positive feedback for the one-week prototype he released to the public back in June 2007.

It was then apparent to him that the vacation wasn't long enough for him to polish it to the level he wanted the game to get to, so he took an extra six months off to work on CRAYON PHYSICS DELUXE. During this time, the game had collected the Seumas McNally Grand Prize in the 2008 IGF competition, so the heat was on for Petri to deliver a stellar product to his fans. That meant another extension to his sabbatical from school; in total he took an 18-month leave from his education just to work on one single project. The happy ending is that it paid off for him, as the game sold very well and was even popular enough to have an iPhone port published by Hudson Soft.



DE BLOB

Developer: Ronimo Games (previously Banana Games)
Web site: www.ronimo-games.com
Institute: The Utrecht School of the Arts (HKU)

Before THQ acquired the rights to this highly rated

THE MISADVENTURES OF P.B. WINTERBOTTOM

Developer: The Odd Gentlemen (www.theoddgentlemen.com)
Web site: www.winterbottomgame.com
Institute: University of Southern California

Sporting the most unique name on this top 10, THE MISADVENTURES OF P.B. WINTERBOTTOM was originally conceived by Matt Korba as his thesis project for the University of Southern California's Interactive Media MFA (Master of Fine Arts) program. Seeing potential in the Flash prototype, Matt and co-creator Paul Bellezza also submitted the game to the Independent Games Festival's student showcase competition in 2009, where a number of publishers took notice of our pie-stealing friend and his amazing time-traveling abilities.

Numerous nominations and awards followed, and P.B. WINTERBOTTOM also made appearances at other shows as well. The game was eventually released on Xbox Live Arcade and Windows by publishing partner 2K Play (a division of 2K Games).

ntgames (that matter)

franchise, DE BLOB was initially a commissioned game project designed by eight students of the Game Design and Development program at The Utrecht School of the Arts and a student of the Game and Media Technology program at Utrecht University in Holland. A prototype which offered about 30 minutes of gameplay was submitted to the student competition in the 2007 Independent Games Festival, and weeks later, it was selected as a student showcase finalist.

After graduation, several members from the team went on to form Ronimo Games and create SWORDS & SOLDIERS, a 2D real-time strategy game that was well received on the WiiWare digital distribution service.

AND YET IT MOVES

Developer: Broken Rules (http://brokenrul.es)
Web site: www.andyetitmoves.net
Institute: Vienna University of Technology, Austria

AND YET IT MOVES is a 2D platformer created by Broken Rules, a development team made up of three former Computer Science students who attended the Design and Assessment of Technology Institute, located at the Vienna University of Technology in Austria. An IGF student showcase winner in 2007, AND YET IT MOVES was also one of the official game selections for the 2008 Indiecade festival. Since then, the game has already made appearances on Windows, Mac, and even the WiiWare platform.

Broken Rules isn't resting on its laurels though, as the company has collaborated with studio radiolaris to create the iPad game GLOBETROTTERS, and has been regularly churning out game releases and prototypes for game development jams just for fun (GAMMA IV and the GLOBAL GAME JAM, to name two).

GEAR

Developer: Team 3
Web site: http://b-lee.net/gear
Institute: DigiPen Institute of Technology

DigiPen has a history of producing some of the best games to be entered in any student contest, and Team 3's win (with the game GEAR) in the non-professional category for GameStop's inaugural Indie Game Challenge competition presented the institute with another award to add to its bulging trophy cabinet. Though the group of five students took over a year to develop the 2D puzzle platformer, everything worked out in the end. Not only did the developers get a sizable amount of prize money for their efforts, they were also offered the opportunity to pitch their game idea to top publishers attending the D.I.C.E. Summit in Las Vegas, where the Indie Game Challenge awards presentation ceremony was held. Quite the reward for a year's work, completed while the team was still pursuing educational coursework at DigiPen.

FRET NICE

Developer: Pieces Interactive
Web site: www.piecesinteractive.se
Institute: University of Skovde, Sweden

The Xbox Live Arcade and Playstation Network versions of FRET NICE were published by Tecmo in February 2010, but the development process for this musical platformer can be traced nearly three years back. The Multimedia Fusion version of FRET NICE was first introduced to the public in 2007, as Marten Bruggermann's college degree thesis project for the University of Skovde, Sweden. The original version is no longer available for download, but the game managed to pick up a Swedish Game Award in the best concept/innovation category when it released.

Pieces Interactive's PUZZLEGEDDON (a 2008 IGF Student Showcase finalist) was also released for XBLA, PSN, Windows and iPhone a couple of months ago, and according to the official site the company is currently working on a new project called KITE.

CARNEYVALE: SHOWTIME

Developer: Singapore-MIT GAMBIT Game Lab
Web site: http://gambit.mit.edu/loadgame/showtime.php
Institute: Singapore-MIT GAMBIT Game Lab

CARNEYVALE: SHOWTIME was created by students from the Singapore-MIT GAMBIT Game Lab—a ragdoll platform game which won Microsoft's 2008 XNA Dream-Build-Play challenge. Shortly after arriving on the Xbox Live Indie Games service in December 2008 (known as Xbox Live Community Games back then), CARNEYVALE: SHOWTIME picked up a nomination for the 2009 IGF Seumas McNally Grand Prize, and was also chosen as one of Penny Arcade's PAX 10 showcase selections that same year.

GAMBIT is quickly becoming a center for the birth of many creative game development projects in the South-East Asia region as IGF main competition entries have begun to come in from Singapore as well.

FEIST

Developers: Adrian Stutz and Florian Faller
Web site: http://playfeist.net
Institute: Zurich University of the Arts, Switzerland

FEIST is a visually striking side-scrolling platformer created using the Unity engine by Adrian Stutz and Florian Faller, two graduates of Zurich University of the Art's game design program. FEIST began as the final thesis for their bachelor's degrees two years ago, and though the game is still in development, it hasn't stopped the duo from picking up accolades and awards along the way. The game first landed wins in the "best visual design" and "best overall game" categories in the 2008 Unity Awards contest, and was also selected as a 2009 IGF main competition finalist, before becoming a student showcase winner.

If the information on the official site is to be believed, then the game could be coming out for Windows and Mac sometime soon.

The visually striking FEIST.

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

Visit our site: www.activision.com

CHRIS DELEON

13 QUESTIONS

hobby video game development

FOR MY FIRST DECADE MAKING VIDEO GAMES, I WAS A HOBBYIST. I CREATED DOZENS OF FREEWARE GAMES WITHOUT ANY INTENTION of making a living from it. Making free games is a great way to get your name out there, and also a good way to pad your resume while you're still working at Best Buy.

As my game making hobby gradually became my profession, I took on a new hobby: helping others get started in hobby video game development. Since then, I have helped dozens of students, from middle school to college-level, begin creating games of their own. A few have gone on to be professionals in the industry, and many are content to just make video games as a hobby.

For some people starting out, there are nagging questions or doubts which can prevent them from getting started, finishing projects, or getting the most out of their games. Here, we'll answer some of those questions about making your first game.

{ STARTING OUT }

QUESTION 1

How can I raise the money I need?

• AAA video game development can cost tens of millions of dollars. Hobby video game development, however, can be done entirely for free. Think of commercial video game development as architecture and construction, and think of hobby video game development as making a pillow fort. If you are spending money on making your pillow fort, you are doing it wrong.

QUESTION 2

What's the best platform, distribution channel, or strategy to make money?

• Has this question ever kept anyone from picking up a basketball, learning guitar, or taking dance classes? When, in rare circumstances, people do wind up doing those things professionally, they don't try it for the first time because someone offered to pay them—they did it for years out of a love of doing it. Overemphasizing money early on can spoil the craft.

QUESTION 3

What if I'm not a good artist, programmer, or designer?

• In that case, you will have more to gain than someone who is already good at those skills. Practical application is a great way to learn. Your initial goal is just to become functionally capable of doing those tasks in order to create something from nothing. In the same way that anyone can draw or sing, even if not incredibly well, anyone can build a level, anyone can make a 2D sprite, and anyone can learn enough programming to at least make something simple. After that, practice will lead to making things that are increasingly complex.

{ INDECISION }

Buridan's Donkey is a parable from philosophy in which a donkey, standing before two equally appetizing bales of hay, was unable to decide on a reason to eat from one pile rather than the other. Paralyzed by indecision, the donkey starved to death, next to more than twice as much food as it needed. In other words, any answer to these next questions is better than no answer.

QUESTION 4

Which programming language should I use?

• If you want to make high-performance downloadable computer games, I recommend C++ (use Allegro, SDL, or DirectX for the graphics/sound/input). If you want to make smaller games for the web so players don't need to mess with installation, I recommend ActionScript 3, which compiles to Flash programs, and can be used for free without Adobe's Flash CS animation tool. Almost all modern console and PC games are programmed in C++, whereas ActionScript 3 projects dominate casual online game sites. I have seen my students get great results from XNA, although that is not something that I have used in my own projects.

QUESTION 5

Which tools should I use?

• I'm partial to free tools that get the job done, especially for a freeware game. I suggest GIMP for 2D images, Blender for 3D models, and Audacity for editing recorded sound effects. In Windows, I program C++ projects using Bloodshed Dev-C++ 5, and ActionScript 3 projects using FlashDevelop.

QUESTION 6

Which part of my dream game should I work on first?

• If it's really your dream game, and you want it made right, don't make it the first project that you work on. I recommend getting a healthy chunk of beginner errors out of your system by making a few very modest and simple projects first. There are two strategies for thinking about scope that I have found effective in guiding initial planning:

"The Console Decades Ladder." Make something of 70s-level complexity first—on the order of PONG, BREAKOUT, or MISSILE COMMAND (if you're feeling fancy). Once you have one or more games of 70s-level complexity behind you, move on to one or two games of 80s-level complexity, then on to 90s, and so on, sticking at whatever phase you prefer.

"The Demo is the Game." If you make a shareware version to show off what the game is about in an effort to rouse excitement about the full product, what needs to be in that demo? Plan on completing the game at that scope as the final draft. If it comes out well, you can build upon it for a longer or more involved follow-up.

QUESTION 7

Are some game types safer than others early on?

• In my experience overseeing and assisting student projects, RPGs and side-view character-based genres are the most common failed or cancelled first-timer projects. This failure is due largely to their hefty art requirements.

Design the game with your art quality and quantity constraints in mind. Some 2D genres—such as overhead racing, side-view flying, abstract puzzle, and overhead space shooters—require virtually no animation. In these cases, rotating and sliding static images via code will suffice. The other major benefit to these game types is that they often do not require hand-designed levels. Spawning opponents semi-randomly, for instance, saves the development effort required to create a level editor, level format, and levels.

Note too that online multiplayer is often much more complicated to build and test than a local-only game. Sticking to single player projects, or games where players can share the same screen and keyboard can greatly simplify technical matters.

{ FINISHING PROJECTS }

Unfinished video game projects are not victimless. They waste the talented work of anyone that helped make content for the project, they cheat players out of an opportunity to explore someone else's imagination, and they cause a severe morale hit to the developer(s). Fortunately, just a few causes tend to be responsible for the overwhelming majority of sunken hobby projects.

QUESTION 8

Are five programmers, five game designers, and five artists enough for one game?

• If nearly everyone involved is a beginner, then that's far too many people for one game. Bigger teams lead to more ambitious plans, longer project schedules, more communication challenges, more development bottlenecks, and more conflicts of both style and opinion.

I've seen student hobby projects with 25 members give up, as the project leader claimed that the team wasn't big enough to finish, while a half dozen teams of 2–4 members all completed their games. Encourage the group of 15 people to split into 5 teams of 3, and you're more likely to see a yield of 3–5 games, instead of none.

QUESTION 9

How can a project be protected from feature creep?

• When the game is in finishing mode, beginning perhaps halfway or three quarters through its scheduled development time, generate a list of the bare minimum that needs to be done before you can consider the game finished. Only allow that list to shrink!

There are two ways to shorten the list: complete an item on the list, or cut it. Whenever you can make a cut without decreasing the quality of the game—even if only because it allocates more time to doing tasks that must be done—make that cut.

QUESTION 10

When should we stop polishing the current idea?

• Iteration, the cycle of tweaking, trying out, and repeating, is an important part of polishing every game. Tuning offers diminishing returns, however, and adds less overall value with each pass. It won't take long before the game is as good as it's ever going to be; if the tweaks are too minor to affect user experience, then it's time to wrap things up in order to advance to the next idea.

The old 90/10 rule of thumb suggests that 90 percent of the player's attention and enjoyment comes from 10 percent of your work. If you can't find something to tune that's likely part of that 10 percent the player will notice or care about, test the game a few more times from start to finish without introducing any new changes, then call the game done.

QUESTION 11

Why does "finished" work keep getting thrown out?

• This happens when a project's decisions are being made in the wrong order, causing everything to change whenever anything changes.

If the main character's jump height is changed late in development, it will potentially break every jumping area in the entire game. Until enemy health, weapon range, and movement are finalized, levels should be throwaway test cases; after those decisions are finalized, and levels are made based on them, those values should not be revisited. Aim to make decisions in an order that minimizes thrashing.

{ WHAT NOW? }

What is one to do with the game after it's done? Plenty, of course!

QUESTION 12

Why aren't more people playing our great game?

• First and foremost, do some marketing. Make sure people know about it. Enter it in a competition, blog and tweet about it, tell your friends, and share it with online communities. No matter how great your game, no one will download it if they don't know what it is or how to get to it.

Assuming you have some visibility on the project, people may be having a hard time getting or running it. In many cases, this happens because the project needs better explanation on the web, or has packaging issues. If it's downloadable, either set up a single-file installer or create a nicely organized zip file with ReadMe.txt/PDF instructions inside it. Try running the game from a few different computers to verify that the package you're giving out to the world isn't assuming a particular library or framework is already installed. Present the game's online link alongside a description plus a few screenshots.

If it's a Flash-based game programmed in ActionScript 3, consider posting to sites like Kongregate or Newgrounds. This often involves little more than creating an icon and writing a short description. After spending months on a game project, spending just a few more evenings making it presentable can have a dramatic impact on the number of people trying the game. This investment is time well spent.

QUESTION 13

Is it possible to know how often people play my game, and how do I retain that audience?

• Keep metrics. Use StatCounter or Google Analytics to keep track of how many people visit your game's web site. You might be surprised which of your projects gathers the most attention. If it's a web game made using Flash, I suggest putting a free MochiBot in the project to keep track of where else on the web the game winds up. Whether you're bragging to friends or updating your resume, it would be nice to know whether your game had 3,300, or 300,000 players.

To retain these players, link back to your site, either via a menu button in the game or through a shortcut in the game's start menu folder. This easy step enables satisfied players to find out more about your current and future work. Even if there isn't much on your site now, there may be in 5 years when you're still pulling in traffic from old games you made in 2010.

CHRIS DELEON *has done the programming, game design, and writing for more than 40 full-featured freeware video games, invented a new experimental micro-project every day for 7 months, and developed many iPhone games (including TOPPLE, BURNIT, and ALICE IN BOMBERLAND). You can find out more about his work at http://deleonic.com.*


Introducing Madison Media Institute's
Game Art & Animation
Associate's Degree
Careers Include:
• Artist
• Animator
• Modeler
• Technical Animator
Program Highlights:
• Motion capture facility
• Creation and integration of assets and animation into the Unreal III Engine
• Utilizing: Maya, Motion Builder, Mudbox, Body Paint, Zbrush, Painter, MEL & Python scripting
Additional Emphasis:
• Story development
• Performance
• Cinematography
• Traditional art
• Color theory
Launch your career today, call us at 800.236.4997
MADISON MEDIA INSTITUTE
College of Media Arts
Madison Media Institute is a specialty college that focuses on media arts including Digital Media Design & Production, Video & Motion Graphics, Media Systems Technology, Game Art & Animation, Entertainment & Media Business, and Recording & Music Technology. The school features over 19,000 square feet of dedicated multimedia classrooms, labs and studio space. Call us today to schedule a personal tour!
2702 Agriculture Drive | Madison, WI 53718 | 800-236-4997 | madisonmedia.edu | admissions@madisonmedia.edu

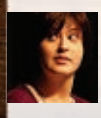
BRANDON SHEFFIELD

the big 6

IN THE LAST FIVE YEARS AS GAME development programs have improved, more and more students have made the leap to professional development. Not just on an individual level either, but a team level, with established companies bringing on entire groups of students, or making a new company of their own. Valve's PORTAL began as the Digipen student project NARBACULAR DROP. Thatgamecompany (FLOW, FLOWER) began life in USC's Interactive Media division, as did The Odd Gentlemen (THE MISADVENTURES OF P.B. WINTERBOTTOM).

In a panel discussion during a Career Summit at the Game Developers Conference, we spoke with developers from all of these companies to find out how they made the leap to professional game development in unique ways, and what pitfalls students should avoid if they want to make it.

Brandon Sheffield: I'm Brandon Sheffield, and we're here to talk about student projects that broke into professional game development. Let's begin with everyone introducing themselves and what they do now, and we'll go on from there.



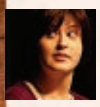
Kim Swift: Hi. My name is Kim Swift. I'm currently a project lead and game designer over at Airtight Games, previously at Valve. My student title that made it big was NARBACULAR DROP, which eventually became PORTAL.

Matt Korba: My name is Matt Korba. I'm the creative director at The Odd Gentlemen. We just released THE MISADVENTURES OF P.B. WINTERBOTTOM for Xbox Live.

Paul Bellezza: I'm Paul Bellezza. I'm producer and janitor at The Odd Gentlemen, and I worked with Matt on P.B. WINTERBOTTOM.

Kellee Santiago: And I'm Kellee Santiago, president and co-founder of thatgamecompany. We had a student project, FLOW, which we took to PlayStation 3, and released FLOWER last year. We'll be four years old in May.

Sheffield: How did everyone's games get picked up? I know that everyone had a different path to success here, but let's talk specifically about your first game getting picked up by a developer or publisher.



Swift: So me, basically, my senior year of DigiPen, which is a game college if you guys have never heard of it, my fellow classmates and I were making a game called NARBACULAR DROP. It was our student project; we had the mindset that we wanted to create something that was a little bit different and get the attention of people, and have something kind of cool to point at in our resumes. Toward the end of my senior year, there was a career fair, and a bunch of developers came and took a look at student projects. Valve, which was actually local to the area, came by, took a look at our student project, thought it was interesting, and brought us in. We demoed NARBACULAR DROP for (Valve president) Gabe Newell, and he apparently thought it was spiffy enough to give us all jobs. So, we immediately started working on PORTAL right after school, adapting our game idea to the Source engine.

Korba: Our game, P.B. WINTERBOTTOM, started off as my thesis project, and we started because we thought it was cool. We didn't really have any plans for what it would do beyond Flash. We ended up crunching for two weeks to get it into the IGF, of course then feverishly submitting updates after that every five minutes. Once they announced the entrance into IGF and had a screenshot, we started getting pings from publishers, which we thought was really weird. We were lucky enough to get into the IGF, so we got more attention and good press and stuff like that. We were getting more attention from publishers, so we decided, "Hey, maybe we should try to do something with this after we graduate."

So after the show, Paul and I had to go back and finish our thesis projects, our papers, and our thesis defense. We were trying to juggle flying around and pitching the game to everybody. Everyone loved it. Some people had different takes on it. They thought it should be in color, or Winterbottom should talk, or there should be dinosaurs in it. Then we submitted to E3. That's kind of what solidified the deal for us because we were winning awards. Our game was still in Flash next to PRINCE OF PERSIA and MIRROR'S EDGE. E3 was really small that year, and so the

A PANEL DISCUSSION WITH FOUR FORMER STUDENTS WHO MADE IT BIG IN GAME DEVELOPMENT

publishers sort of came around again, and they were like, "Oh crap, we better figure out this space because someone else is going to buy this game." Then we started the professional game in October of 2008 and worked on it for a year.

Santiago: Yeah, I think Kim and I graduated ... Did you graduate in 2006 as well or ...?

Swift: 2005.

Santiago: 2005. Yeah, so it was around the same time. That was when there were student projects coming out like NARBACULAR DROP, which were just kind of awesome. And publishers were seeing it but didn't know what to do with us. They were like, "Wow, we really love this game, but you guys are just students. You've never shipped a commercial title. We don't know what to do."

So how it started for us is Jenova [Chen] and I collaborated on the student game CLOUD at the USC Interactive Media division. It was done under a grant from EA. Jenova had arranged with the people who were sort of the ambassadors to USC to do a pitch basically as an educational experience. We thought we'd get great feedback and it'd be better than a job interview. You know, we'd have this undivided attention where we could talk to them. And Jenova and I are both kind of "go big or go home" people, so we created this pitch that was as professional as we thought we could be without having much experience at it.

We actually opened with this 15-minute section on how we wanted to change the game industry and make these new emotional games that we thought would appeal to new audiences. Then we talked about the student project, and about an idea we had for a next-gen console version of it. Through that process, EA was actually the one that gave us feedback to look into the digital distribution space. And we looked into it right after XBLA had launched. PSN and WiiWare weren't even announced yet. But we had a sense that everyone was going there. We thought, "Wow, this is actually starting to seem like a feasible way to make a living, to keep doing the games that we wanted to do." Up until that point, we really thought, "Well, we're going into the industry. Put in the three to five years at a large studio, save some money, make some contacts, and then we'll regroup and start our own company." And suddenly it seemed like, "Wow, we can actually do this sooner than we thought."

I think that attitude, the way we structured the pitch and everything, is what led us into the three-game deal we ultimately got with Sony Santa Monica, because they saw something beyond just the game we were talking about. In fact, we changed from working on the game we were talking about to FLOW, primarily because the design of it was more fleshed out. We thought as a first game endeavor, it would be easier. It turned out to still be super, super hard. But we had a better first prototype of it.

Sheffield: What kind of tips might you have for building and motivating a team while you're at school? I'm talking about keeping people together and making them actually do the work that you want them to do.

Bellezza: First of all, when recruiting a team, we kind of just had an open session. Actually, we were talking to Jenova one day about how he was working on his team when he was at USC, and he pretty much said, "You know, just hold meetings, and whoever shows up after the second or third week, that's your team." [laughs] And that's what happened. We had a dedicated team of designers working with Matt and myself on building the game. And they were super motivated. I think part of that is because of the design of the game ... We had prototypes of the recording mechanics in WINTERBOTTOM, and we had the character, but there were still a lot of holes in the design. There were enough holes that other people could contribute to it, to make their own stamp, and that was really motivating for other team members. Everyone was included.

So, what we would do is we would have a little thesis of what we wanted to hit, like whether it was a blue-sky week or whether we wanted to explore a particular mechanic. Everyone would go and design for a couple days, come back, and talk about it as a group. And doing that a lot kept everyone really, really involved, moreso than me just buying everyone pizza on my credit card. So, that's kind of one way that we approached that.

Swift: I think that collaborative design is incredibly important. It keeps people invested in the project. NARBACULAR DROP, which became PORTAL, we designed by committee. There wasn't just one person saying, "Oh, this

KIM SWIFT PHOTO BY VINCENT DIAMANTE

is exactly how we're designing this game." We all sat down at a whiteboard—artists, programmers, me as a designer—and we actually made decisions as a group. You get not only that emotional investment in the game, you actually build a better product because you get a wide variety of opinions on a topic. You would think that, "Oh, you're just going to get into arguments and stuff all the time," but actually, you put stuff up on a whiteboard and you're all working toward the same goal. The logical conclusion of which way to go tends to shake out pretty quickly.

Sheffield: If you're going to form a company, or even if your whole team, as Kim's did, kind of moves into a larger developer, how do you prepare to tackle the business end? Obviously, a lot of the people making the games are going to be on the creative side, and it's really hard to make that leap to thinking, "What do we need to know about contracts?" or "How are we going to get paid?"

Korba: We have a lawyer. That was one of the first things that we knew that we would need when we were getting these contracts. That's a big help. As far as the business dev side goes, we just had to learn. I think we got our company business account like two days before we started, because we realized that we had a check that we needed to deposit, and we couldn't just put it into our own personal accounts. There's a lot of stuff that's just obvious that nobody tells you. Like, yeah, you should probably buy a mop to clean that office.

It's a lot of trial by fire. Even though we talked to thatgamecompany, and we had talked to others that ran their own studio and got great advice, there was still a lot of stuff that you have to figure out on your own. We still mess up all the time and are learning new things every week and trying to get better, especially after only doing this for a year now. But we just try to get advice from as many as people as possible. Services like ADP for payroll and everything else, you know, we have to wear many hats. Like Paul said, he's the janitor. He's actually not joking.

Santiago: Yeah, this is definitely one of those areas where it helps to be really nice to everyone you meet while you're in school and while you're at these conferences because it is a trial by fire, and the way we did it was by basically building up our own community of personal advisors who could help us out when we had questions. And that came from our faculty at USC. It came from people who came in to lectures at the school, and then also other independent developers who ... I find games to be a great place to work in because everyone is pretty helpful and if you can identify kindred spirits, then everyone really pitches in to help each other out and get through these things.

Sheffield: I've also heard advice that you could check out if your school has a business division and talk to people there. But you have to be very careful to make sure that they're on the same page as you and not out to screw you or something. There are business guys out there that really care about the creative end. Like if you look at The Behemoth, who made CASTLE CRASHERS and all that, the guy who's at the head of it, John Baez, he's a really great businessman, but he really, really cares. So, you've got to find John Baez before he's John Baez.

Santiago: Yeah, well he is one of the people I call. [laughs]

Sheffield: Kellee mentioned professors—how much help can you expect or ask from professors when you're making a student game and forming a team, and when you're trying to figure out what platform you may use like Game Maker or Flash?

Santiago: At least from a project perspective, one point of advice I have is that it's really important that you're very focused when you ask for help. I'm not a professor, but for instance, I'll get an email every now and again saying, "Hey, I'm interested in making games. How do you do it?" I really want to help everyone, but that's such an open-ended question. I don't have time to have that conversation. Professors are the same, especially in the game design programs now because they're just filled to the brim with students with awesome ideas who are super motivated to do great games. I think it helps, when you go for that advice, to be really focused on what it is that you need from them, and make sure you're really leveraging the time you have with them.

Korba: I just have an hour therapy session with my professors. Professors are always busy, but if you can have an hour a week or something with your professors, a lot of good can come out of that. Tracy (Fullerton), who's actually sitting somewhere in this room ... Hour therapy sessions with her are really what helped me a lot. When the project started, I didn't know how to code in Flash. I didn't know how I was going to do this recording (of time) thing, but just through talking, it was like, "Well, why don't you just fake it? Why don't you just have an animation play and interact with that and see if it's even fun, see if the idea of you playing something is even fun?" Just stuff like that, talking about how to prototype stuff and how to get your ideas out.

I used my professors' help for guidance in that area a lot. Then of course, it helps to have someone that's gone through this and had experience with getting a game to the IGF as a student or bringing a game to a service. We actually were lucky because we had thatgamecompany to look at and see how they did it. The mistake we made, of course, is that just because one person does it one way doesn't mean it's always going to work the same way. Everyone's going to have a different path. So, when you are getting advice and you are talking to other people, just realize there's not one way to do it. It's not going to work the same way. It's probably going to work a different, crazier way.

Swift: For me personally, being at DigiPen, I was extremely fortunate to have professors that were supportive and promoted student projects to local game companies. It's definitely worth looking into, finding someone that can be a mentor and sort of have your back, regardless of whether they're a professor or an industry professional. As people have stated before, help is good. You'll need it. And be grateful for it.

Sheffield: Talking of not knowing Flash to begin with but then winding up using it, what would you all say is your recommended platform for people getting started? My impression is that there are a lot of designers and artists in these school programs, but coders are kind of harder to get a hold of. So, should you do it yourself or should you try and get someone from the computer science division? And if you do want to do it yourself, what platform do you recommend?

Swift: So, I think before you actually go into the industry, no matter if you're a programmer or you're an artist or you're a designer, you need to work with other people outside of your sphere because that's what you're going to encounter in the game industry. You're not going to go to a game company where they're all artists. You know, there need to be coders there. There need to be designers there. The more experience you can get working with

PORTAL began life as student game NARBACULAR DROP

different types of people, the better. I would highly recommend it.

Korba: For me, I didn't know any code before I started, and I had the attitude that it doesn't matter what the platform is as long as the idea comes across. I think a lot of times, in school, people get caught up with the technical aspect of it. "I can't code. This has to be some giant thing in C++." I used Flash and worked with that, and I didn't even know Flash. I learned it step by step. I would go on the forums and be like, "Okay, here's how you make a platformer. Copy and paste." Someone told me what an array was, and then I figured out you could store information in that and use that to do the recording system. I had no idea. I think, whether it's Game Maker or Unity or whatever, as long as you can get your ideas across ...

For me, it was important to have something before I got my team just so everyone was on the same page, everyone could sort of see what we were going for because I think a lot of times, especially with student designers, we have lots of crazy ideas. To find people to work on them and spend their own time on it, they need to get behind something that they can see. The other thing is if you're an artist and you can't code at all, you can do slideshow mock-ups. You can use a PowerPoint presentation to get an idea across. That's how I started out. What I could do is I could model in Maya, so I made really nice renders of WINTERBOTTOM and a comic book step by step of how the mechanic would work, which totally wasn't anything close to what it wound up being, but it got people excited and it got an idea out. So, whether it's paper prototyping or whatever, use whatever tools you have to get the idea across the fastest and easiest way.

Sheffield: How do you differentiate yourself as a student game? As we've mentioned, there are a lot now, and publishers are noticing some of these. How do you make your game really be the one that people look at?

Santiago: Certainly, if you look at IndieCade and at the Independent Gaming Festival now, every year, the bar is getting raised on the level of polish in both festival games and also what's coming out of schools. I think that's a new standard that we're seeing be set as a way of standing out, just having a polished project. For those of you that were at the Independent Gaming Festival awards, everything that was nominated for visual art this year is 2D, so it doesn't mean that you have to have a really high level of production in your graphics or in the tuning of your game, it just all has to be at the same level. That's something that at thatgamecompany, that when we were students, we tried to aspire to.

We compared it to the barrel with the planks holding in water. The highest level the water can go to is the level of the lowest plank. So, you can have great graphics, you can have great gameplay, but if your engine is busted, then it's not going to stand on its own. If everything is sort of at the same level, I think that, to me, presents a tight experience. And the second is, being an IGF judge myself, don't go for the "me too" when you're thinking of projects that you want to possibly get a job on. I think those kinds of projects are great to start off to teach you how to make a game, because the game design is already done ... I think it's just a real pain in the ass to make your own game. So, as long as you're putting all that blood, sweat, and tears into it, you might as well do something that's a little different.

Korba: The question was how to make a game stand out. Silent film. Delicious pie. Time travel. Works every time. (He is referencing WINTERBOTTOM here.) I think you just have to do something personal, and this is kind of what Kellee was saying, too. Whatever makes you unique and your own personal experience, use that. A lot of times, in student films, people reference other films, and it's becoming the same way in games. It's like people are referencing a lot of other games. But if you take something that's outside of games in your life or your experience and put that in, it's always going to be unique because nobody else has that experience besides you.

The other thing, just as a more technical thing to make your game stand out, especially for the IGF, you have to take into account how your game is going to be judged. There are tons of games, there are judges that have to judge a bunch of games. You want to grab them in the first five minutes. Whether or not your game is a quick thing, try and structure it to grab people fast. That's really what's going to get people involved in the game. I would love to sit there and play all these games for hours and hours and hours and get to the cool part, but unfortunately like we're all busy. We just don't have time.

Also, realize that on the show floor, people are walking by screens and deciding what they think is interesting based on a quick glance. So, if you have something, you should structure it, your demo or whatever it is, to be like a demo, like a trial on Xbox Live. Get to the core experience fast so people that are walking by can see your game, because most people will stay for five minutes and find out if they're interested or not. Like, "What's cool here? I'm going to go around and check all the games." Make something that grabs and grips pretty quickly.

Swift: I totally agree with Kellee in terms of having a level of polish to your game, and for you to get that polish, do not bite off more than you can chew. It's incredibly tempting to go, "Oh, I'm going to make this and this and this

and this." No. One idea. Focus. The more time and energy you can put into that one thing and polish the hell out of it will make for a better experience. You'll get that grab in the first five minutes.

Sheffield: Yeah, all these people here, their games that wound up doing well had a very strong sense of an auteur behind them. You can definitely tell that this game was not made by somebody else. It was made by this specific group of people. It's like a film auteur. It's like a similar concept. So, Matt, you were talking about the kiosks and people walking by. If your school has a career fair, and this applies to Kim as well, how should you get noticed? What should you do to grab people? Like, if there are demos in kiosks or perhaps if there are not, what can you do to grab the attention of these people?

Swift: Literally, grab their attention. If people are walking by and they have just a passing glance, and you can see they're kind of looking, go up and talk to them and say, "Oh, hey, are you interested? Come and play." Sell yourself and sell your game. Don't just stand there and watch. Grab them over, pull them over, and have them play your game. For us, Valve took a look at our game in a quick passing glance and didn't really think much of it because it's like, "Oh, wow. It's very brown." But beyond that, we had a ... Once again, fortunate to have really supportive professors. One of our professors actually grabbed the representatives from Valve and dragged them over and had them play, and that's what convinced them to bring us in and develop in the first place. So, drag people over to your screen, sit them down, have them play.

Korba: I think if you're at a career fair or IGF pod or whatever, you need to be there full time. If you get a game in the IGF, you never know who's going to walk by, so make sure you're there. Also, same with team members. We had a three-pronged attack. We always had someone playing the game, whether it was (an attendee) that was actually playing the game or one of us. We had postcards that had cool art on it, and so Paul would just go out, spam everyone, and pass those out to everyone and really just say, "Have you seen this?" We usually had a big crowd because we had something that was pretty crazy looking to begin with but also because Paul just handed out a crap-load of flyers to everyone that was in reach. Then, I would actually explain the game to people when they came by as it was being demoed. Even though we had something that people were interested in, we still had to hustle to get people to actually play it.

Santiago: Yeah, I completely agree with that. It definitely helped that CLOUD was primarily light blue. That stood out a lot and sort of acted as a gravity field for us for people walking by. We also divided up the time, making sure that someone was always there playing the game, and if you are in a situation where you think you won't be able to be there all the time, it does help to build what's called in the industry an "attract mode" in your game. So, if you leave it, if you put it on pause, it shows screen stills of the game or it plays itself somehow or stuff like that to show people, so you don't just have this waiting screen that's sitting there, and people probably won't check it out.

And then if you don't have someone like Paul on your team, something that we did was basically write out what the game is, how we wanted to articulate the main points of our game, in a concise, quick manner. So, if someone just had one minute to stop by ... Some people call it the elevator pitch, that quick way to clearly articulate what it is, why we're here, what we're interested in, whatever it is you want to get across while you have that person's attention. We wrote it down, we bounced it off of people, we refined it, and then we all knew what we had to say. Maybe the first couple of times it sounds like you're reciting something you wrote down, and then it becomes natural. And then it's great because everyone's sort of communicating the same thing, and you've prepared it ahead of time so you're not really taken off guard.

Even now, we'll try and guess what people are going to ask of us when we're demoing a game and write it out beforehand. Again, we always know what we want to say because yeah, I'm not quite as witty as Paul is, so I have to do that.

Sheffield: Is going to school near a development hub helpful and important, and if you can't go to school near a development hub, any tips for that? Because I know that one guy on the FLOW team, he made his first big game in the middle of nowhere with no support.

Santiago: That's true. Our lead engineer, John Edwards, who was the first person to join us after Jenova and I, we met him through the former Slamdance Guerrilla Game Maker competition, which no longer exists, and the IGF. He was here with a game called OCULAR INK, and it was really cool. It was a situation where he had developed this game away from any sort of hub of other developers. He certainly had been a part of an online development community, TIGSource and resources like that. He came to conferences to connect with other developers.

I think that's an important thing to build into your yearly schedules: make sure you can meet with people face to face. Now, there are so many small development communities. You can probably reach out in your local community and put flyers up and see if people want to get together to talk about games; that can lead to doing some game jams. Global Game Jam is another great opportunity. I think

Thatgamecompany's FLOWER in attract mode.

there are a lot of opportunities now, especially through all the Internet communities surrounding game development.

Korba: Yeah, I think for us, it helps to be in L.A. with a lot of developers to get advice from, where we can go and check out their studios, and that kind of thing. But the cool part about games is you can do it wherever. Like, if we wanted to move our studio to Maine, I think it would be alright because all the communication is done over the internet anyway. I think the trick is what Kellee was saying. You need to come to these sorts of things to meet other people and touch base. Whether you put your game up for free in Flash or submit it to the IGF, there are a bunch of ways to get noticed now. But I think going to school in an area where there is a dev community is definitely a benefit. I think there are, again, many different paths.

Sheffield: What kinds of mistakes did you all make that you would advise other people against?

Korba: Before WINTERBOTTOM, I worked on an MMO played over BitTorrent, a dream battling game. That never made it anywhere. I think the problem with that was just scope. We made mistakes where, in school, before WINTERBOTTOM, we had these game ideas that were just so ridiculous that there was just no way we could pull it off. I think it's important to look at your team and look at yourself and do what you're good at. And what I was good at was not making an MMO over BitTorrent. Lots of people can do crazier things than others and bigger things than others. It's important to look at what you actually can pull off.

With WINTERBOTTOM, there was so much crazy stuff in it that I decided to do it as a 2D sidescroller because I knew that was an easy thing to build upon. Then, there turned out to be other design reasons why that was a good choice, but initially, it was like, "Alright, let's do something that's already a solved problem to put this crazy recording system on top of."

Santiago: God, I mean, we made so many mistakes. [laughs] It's really hard to pick out one. But I guess one of the main ones was probably feeling like our game making would conquer all. There are some really fundamental needs when you're working on a game team for leadership and collaboration skills, which none of us had focused on nearly enough while we were in school. Actually, from what I hear, DigiPen provides great opportunities because everyone's on very similar schedules, right? [Swift nods] So, you're working together on full-time schedules, whereas most schools, USC being no exception, everyone has different schedules, and so you're sort of working in the pockets in between. And then to take that to a full-time development schedule was a really rough transition for us. It was a really great learning process that I've talked and will talk much more about at GDCs. So, just really ... I think we needed more guidance. Read more, learn more, and understand more about leading teams and management skills.

Swift: Be aware of your constraints, first and foremost. I basically worked with the same team for over four years at DigiPen. Our first couple of games, yeah, not so good. You are not making a JRPG in your time at school. Just get that out of your mind right away. I definitely watched students go down that path, "Oh, we're going to make this cool JRPG with so much dialogue and blah, blah, blah, blah." Just be really realistic with how much you can accomplish. I mean, if you are going to school, you have other classes. You have other projects.

Sheffield: A lot of students will jump at their first chance to work with any developer that they can get a job with, in any position. Even if they want to really be an artist, they'll say, "Okay. I'm going to get this QA job at EA, and I'm totally going to work my way up." What would you say to them?

Swift: So, if you have a dream job in the industry, I personally would say, "Try and go for that." Yeah, sometimes it's good to get your foot in the door, but not all testers get promoted to jobs in actual development. And so, if you really, really want to be an artist, go to interviews and put your portfolio out there and have people critique it and tell you what's missing or that you're not selling yourself enough. Then work on that.

Korba: I went to film school before I was doing the Interactive Media program at USC, and I think the mistake there is a lot of people think that they're just going to go through film school, and then Steven Spielberg is going to come to their graduation ceremony and hand them the next big movie. It just doesn't happen like that. School is a great time to be in a safety net to create something. So, that's what you should do. You should make something. If you want to be an artist, you should make art. If you want to be a designer, you should design things. Nobody is going to give you a job for something you haven't already done. So, you should really use school as a time to do something and show it off. I think a lot of the time that's how people get stuck in this path of "I'm going to try to go here because there's an opportunity but then work my way up there." It's because they might not have something to show why they should be in this other position or what have you.

Bellezza: I would say really look at the companies whose culture vibes most with your design sensibilities and philosophies about game making because if you apply to a place and it's not quite to your liking, it's okay—if you need to get a job, get a job, but always have an end goal in place. Know where you want to go, and build up the skills that you need to get there or go to GDC and meet the right people and network to eventually get there. My biggest advice is I've seen people burn out and kind of stagnate. They had big dreams. Then they got a really hard job in an art department where they weren't appreciated, and they were done. I think you've got to kind of build off of that and keep your eye on the long-term prize of where you want to be, who you want to work with, and you'll get there.

Santiago: Yeah. I couldn't agree more. The more focused you can be in your goals, the better off you will be in the long-term. I know plenty of people who just want to work in games, and in that case, then go for it. If you're really thinking "I just want to be in it," that's fantastic. There are a lot of jobs we need people like that for. On the other hand, if you know you want to be this kind of artist ... I mean, art is an example that's easy to keep going back to because you have that portfolio. But it's the same with programming. If you can identify that "these are the kind of games I want to make," you start building out your portfolio to reflect that.

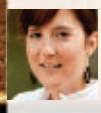
Simultaneously, you need to start searching out those teams, those projects, and those companies that are in line with the style of game developer you want to be. I actually met an environment artist on the GOD OF WAR III team, which is down the street from us, who was a student. He knew he wanted to make games that were different. He really liked our style of games. That's how we got connected with each other. But he wasn't seeing opportunities at studios like ours. He had seen the documentary on the GOD OF WAR team on the GOD OF WAR II disc, and he really wanted to work there based on that, based on just getting to know them through that documentary. He said, "That's the kind of

culture I would like to be a part of before I get to work on the kind of games I want to make."

He went to an extremely violent game, but it was in an environment ... And the art team there is just unbelievable. So, it was just a great opportunity. However, right off the bat, they told him, "We don't hire students." But he had his mind set on it. So, he kept going after it. He basically got presented a situation—I think he was at a conference like this. He met one of the leads in the team, and because he had set his mind on that, he could start a conversation, "Oh man, I would really love to work for you guys." And this person, since they kind of got to know each other in person in a more casual context, said, "You know, why don't you just send your stuff to me. I'll take a look at it and at least give you feedback." And long story short, that led to him getting onto the team. So, even though off the bat, he had been totally denied, because he really focused his energy in that direction, he was able to get a position with that team.

Sheffield: As a closer, here are a few things that you can do to get noticed a little bit—submit articles to GameCareerGuide.com, do postmortems of your game, and put them somewhere, even if it's just on a forum or your own site. Join forums like TIGSource and other indie game hubs. Just meet with people and try to get your name and your game out there as much as you can. We're going to open it up to audience questions now.

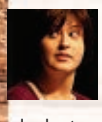
Question 1: *This is a question for Kellee. I was wondering if you could give us your elevator pitch for* FLOWER *because I imagine that just saying you're the wind and you're blowing a flower doesn't cut it, so how did you do it?*



Santiago: So, FLOWER is our attempt at creating a video game version of a poem. You're presented with an urban apartment initially with a dingy gray scene, and there's a single flower on the windowsill. As you focus in, you're taken into a dream in which you're blowing wind across serene, imaginary landscapes, touching flowers and impacting the world as you go. [laughs] That was a hard one for us, that one.

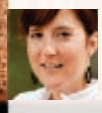
Question 2: *This is kind of a two parter about education. I'm wondering, first, what you think the value of a degree is, or Bachelor's versus Master's versus just being able to demonstrate your skills and talent. And the second part is I'm wondering if you guys think, in this indie space, if there are positions where you can have input on programming and design versus really needing to specialize on one or the other.*

Bellezza: Basically, if you're in school and you're getting your degree ... I think the best thing to know is know the skills you want to develop and take classes that are going to build that. I mean, it's an obvious answer, but it's important. My undergrad was in psychology. I use a lot of that in my day to day, but it wasn't until grad school where I started picking up a little bit of scripting and Photoshop to be able to help out on a game team. So, just building those skills are important. Do you have to have a degree? If you have the skills to pay the bills, not necessarily, but go to school and learn what you need to learn.



Swift: Basically, for me, what's good about having a degree is it's showing a level of commitment to finishing something. When people look at your portfolio and resume, that's what they want to see. Are you a finisher? Do you take a commitment seriously? Because that's what you're going to need to create a game. There's going to be hard crunch times. It's not all song and dance and happiness. If you've been making games on your own and you've "shipped" titles, then that's what people are really going to be looking for in your resume.

Question 3: *I have a question for thatgamecompany specifically. In the transitional period before you got established as a real organization, were there any hairy situations in terms of supporting yourself? Did any of you guys have to move back into your parent's basement or live on pizza or something like that?*



Santiago: Yeah. We had really set our mind on launching FLOW in the launch window of the PlayStation 3. The reason that I bring that up is it put us in a situation where we needed to start developing our game before our contract was finished negotiating. This isn't an uncommon practice in game development, but you can't get paid until that contract is signed. There are some ways around it. You can do the letter of intent, you know, small placement stuff. So, we were basically ... For the most part, we were paid on FLOW the day that it shipped. [laughs] And so, we were all in a really hairy situation. I actually had to borrow a small amount of money from a relative just to get by. I mean, it helped that it's like, "Yes, we're doing this. This game is launching. I just need it for a few months." But yeah, that was the lowest point for us financially.

Question 4: *I currently go to an art institute. My team production, which is our final game, we finish next quarter. It's a year-long project. In Las Vegas or Nevada, there are one or two different game companies, one of which is a triple A kind of place. Considering there's only one big game company in Vegas and our game is not their style, how would we pitch our idea to them or to other companies? Just mail them a copy of our disc or because we're using Unity we can run if off a web site? How would you send your game to other companies outside of Nevada?*

Bellezza: Submit it to every festival that you possibly can (see Competitive Spirit on pg. 62). Get it into everything and promote the hell out of it. Also, you can go to IGDA meetings, if there's a chapter in Vegas. Any gathering of game developers ... Also, you're in Vegas. You can drive down to L.A. for E3 or something. If you can get to a hub and network with other people, that's how you're going to do it. You can throw it into the void, and you can build up your community and support it, but I think the best and strongest thing for students is at every festival, no matter how small, submit. It will help.

Question 5: *For the* WINTERBOTTOM *guys. I was reading a Game Informer feature on you. Which one of you ... Maybe my facts aren't straight, but which one of you was actually teaching Biology before you got into this. That was a big inspiration man, just because, I felt like, oh, I'm student teaching in the fall, and once I do that, I'm never going to be able to think about video games again.*

The thing is, you guys were talking about being a part of game developing communities and being part of university communities and stuff like that. What if you're outside the university communities and you want to crash that to find people like you guys? What's a good way to do that without having to use the Internet because it's so impersonal? Do you just go there and talk to professors? Are they just open to you?

Bellezza: So that was me. I quit teaching and became a QA tester. I was determined to go back to school and learn game design. I pinged one of the professors, Tracy Fullerton at USC, like once every month. I looked at their blog and looked for board game night or whatever, and I just kept on showing up. [laughs] Then I applied, and I talked daily ... Well not daily. I talked a lot. I just busted in there, and eventually, when I applied, they knew who I was, and they saw.

gamedeveloper

GAMASUTRA
The Art & Business of Making Games

GAME CAREER GUIDE

FG FINGER GAMING

indiegames.com

GAMES ON DECK

WORLDS IN MOTION.BIZ

GAME SET WATCH

GAMERBYTES

SERIOUS GAMES SOURCE

For more information on our products and services visit:

www.jointhegamenetwork.com

UBM TechWeb

PHOTOS BY VINCENT DIAMANTE

Independent Games Festival Student Showcase Competition 2010.

Competitive Spirit

ENTERING INDIE COMPETITIONS CAN HELP YOU MAKE A NAME FOR YOURSELF

TIM W. BOON

If you've read the IGNEOUS postmortem on Pg. 32, you're already aware of how important student game competitions can be. Becoming a finalist in the IGF student showcase or one of the company-specific competitions can help jump-start your career. To that end, we've compiled a number of competitions students can enter, along with their fees and restrictions. These dates can also help keep your project focused on a specific goal or timeline. Without further ado, here are the important competitions for 2010 and 2011.

INDEPENDENT GAMES FESTIVAL STUDENT SHOWCASE COMPETITION

WEB SITE: www.igf.com
SUBMISSION DATES: July to November 2010 (exact dates to come)
ENTRY REQUIREMENTS: Open to all student game developers worldwide, entrants must be at least 13 years old.
ENTRY FEE: None
ORGANIZERS, SPONSORS: UBM TechWeb
/// An award for the best student games. Student mods to existing games can also be submitted. Ten Student Showcase winners are selected to be showcased in the IGF Pavilion at the Game Developers Conference, and each winner receives a $500 travel stipend. One game will be selected for the Best Student Game award ($2,500 prize), to be revealed at the IGF Awards Ceremony.

INDIE GAME CHALLENGE

WEB SITE: www.indiegamechallenge.com
SUBMISSION DATES: May 3rd through Oct 1st, 2010
ENTRY REQUIREMENTS: Open to all individuals and teams worldwide (residents of U.S.-embargoed countries are ineligible to participate), entrants must be at least 13 years old.
ENTRY FEE: $100 (fee is waived for residents of certain states in the U.S.)
ORGANIZERS, SPONSORS: The Academy of Interactive Arts and Science (AIAS), GameStop Corp., and The Guildhall at Southern Methodist University (SMU)
/// A game development competition that presents students with a chance to get their games published and win up to $100,000 in the Non-Professional category. Each team member (up to five members in a team) can receive a $500 travel stipend to attend an awards ceremony at D.I.C.E. 2011. The finalists will also be given the opportunity to present their games to interested publishers during the event for possible publishing contract consideration.

DREAM-BUILD-PLAY CHALLENGE

WEB SITE: www.dreambuildplay.com
SUBMISSION DATES: Annual, 2011 dates to be announced
ENTRY REQUIREMENTS: Submitted game must be in English and playable on an Xbox 360 console. The competition is open to all students ages 16 and above who reside in qualifying

regions.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Microsoft

/// Microsoft's Dream-Build-Play Challenge offers up to $70,000 to be split among four selected winners, and each of them will receive an invitation to release their title on the Xbox Live Arcade service. An Xbox 360 game demo must be submitted at the official contest site to qualify for the competition.

UNITY AWARDS

WEB SITE: http://unity3d.com/contest

SUBMISSION DATES: Annual, 2010 dates to be announced

ENTRY REQUIREMENTS: Game must be created using Unity. Entrants must be at least 18 years old.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Unity Technologies

/// The Unity Awards game development contest is an annual event that honors the best games created with Unity. There are four categories to compete in: Best Overall Game ($5,000), Best iPhone Game ($2,500), Best Technical Achievement ($1,000), and Best Visual Design ($1,000).

LEVEL UP (INTEL VISUAL ADRENALINE GAME DEMO CHALLENGE)

WEB SITE: http://software.intel.com/en-us/contests/levelup2010/contests.php

SUBMISSION DATES: Annual, 2011 dates to be announced

ENTRY REQUIREMENTS: Open to participants worldwide. Entrants must be at least 18 years old.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Intel Corporation

/// Level Up is a worldwide competition that provides developers greater exposure to the gaming community and a chance to win prizes based on the utilization of the features of Intel-based platforms. The three categories to compete in are: Best Game for a Desktop, Best Game for a Laptop, and Best Game for a Netbook. The winner of each category will be awarded $1,000, among other Intel-sponsored prizes.

INDIECADE

WEB SITE: www.indiecade.com

SUBMISSION DATES: Annual, 2011 dates to be announced

ENTRY REQUIREMENTS: Submitted game must not have funding from a major publisher, no age requirement.

ENTRY FEE: $45, or a discounted entry fee of $35 for early submissions

ORGANIZERS, SPONSORS: Creative Media Collaborative

/// IndieCade organizes a series of international events to help developers showcase their independent games, held across multiple venues throughout downtown Culver City, California. Selected entries will also be presented at E3 and IndieCade Europe in the UK during Gamecity. IndieCade welcomes student games and games developed by universities, schools, and non-profit organizations. Works in progress are permitted and encouraged, but they should include at least one finished playable level.

SENSE OF WONDER NIGHT

WEB SITE: http://tgs.cesa.or.jp/sown/en

SUBMISSION DATES: April 20th through July 11th, 2010

ENTRY REQUIREMENTS: Open to all individuals regardless of nationality, age, or occupation.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Computer Entertainment Supplier's Association (CESA), Nikkei Business Publications, Inc., IGDA Japan

/// Sense of Wonder Night selects a group of indie developers from around the world to demonstrate their game or prototype to an audience at the Tokyo Game Show for 10 minutes. A SOWN Pavilion will be set up at the TGS show floor so that developers can also present their games to the event attendees.

PAX 10 (SEATTLE AND BOSTON)

WEB SITE: www.paxsite.com/paxprime/pax10.php, www.brownpapertickets.com/event/89226

SUBMISSION DATES: Annual, 2011 dates to be announced

ENTRY REQUIREMENTS: Open to all independent developers without a traditional publisher-developer relationship in place. Must be able to demo the game live at PAX.

ENTRY FEE: $50

ORGANIZERS, SPONSORS: Penny Arcade

/// Penny Arcade chooses the 10 best games from all submissions received for their annual PAX 10 competition, and the developers of those games will receive invitations to showcase their work in a special booth at the Penny Arcade Expo consumer event.

IndieCade 2009.

INDEPENDENT GAME DEVELOPERS' COMPETITION

WEB SITE: www.indiepubgames.com/news.php?id=2

SUBMISSION DATES: May 1st through July 31st, 2010

ENTRY REQUIREMENTS: Open to individuals who are at least 18 years old. Students are eligible as long as their college or university does not own what they create.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Zoo Publishing, Inc.

/// The Grand Prize winner for IndiePub's Independent Game Developers' Competition gets a $100,000 cash prize and the option for a publishing deal with Zoo Games. The Community Favorite category winner receives $5,000, and each of the five Public Recognition Winners in the categories of Technical Excellence, Art, Audio, Design, and Staff Pick receives a $1,000 prize.

YOYO GAMES COMPETITION

WEB SITE: http://glog.yoyogames.com

SUBMISSION DATES: May 14th through August 31st, 2010 (for Competition 06)

ENTRY REQUIREMENTS: Games must be made using Game Maker 7 or Game Maker 8.

ENTRY FEE: None

ORGANIZERS, SPONSORS: YoYo Games

/// A semi-regular competition for Game Maker engine users. Participants must create and submit a game that incorporates the announced theme. The chosen theme for Competition 06 is "Discovery." First prize is $1,000, the runner-up gets $500, and the third-place entrant receives $250, all paid via PayPal.

GAMMA 5

WEB SITE: www.kokoromi.org

SUBMISSION DATES: Annual, 2011 dates to be announced

ENTRY REQUIREMENTS: Game must incorporate announced showcase theme.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Kokoromi

/// Gamma showcases are free for anyone to enter. Chosen games will be showcased at a public party, and developers are given about six to eight weeks to create a game between the time of the theme announcement and the final submission date.

JAYISGAMES' CASUAL GAMEPLAY DESIGN COMPETITION

WEB SITE: http://jayisgames.com/tag/competition

SUBMISSION DATES: CGDC #8 dates to be announced

ENTRY REQUIREMENTS: Game must incorporate announced theme, entries must be specifically created for competition.

ENTRY FEE: None

ORGANIZERS, SPONSORS: Jay is Games

/// A regularly held competition for Flash (or browser-based) game developers to participate in. Prizes vary, with the competition winner usually receiving a $1,000 cash

Student Project Disasters!

HOW TO TURN KNOWLEDGE FROM FAILED GAMES INTO A BOON FOR YOUR CAREER

IF YOU'RE IN SCHOOL TO learn how to make games, it's a pretty safe bet that one of the student games you'll work on could be charitably described as a complete disaster. Though the sting of defeat may take a while to subside, it's useful to remember that difficult projects can be a much more valuable learning experience than the ones that go smoothly.

The experience of a "failed" game can even give you an advantage in job interviews– if you know how to approach the topic. As Darius Kazemi, a game developer who has mentored students at Full Sail University explains, "In almost every interview you'll ever go to, you'll get asked about a time when something went wrong with a project and how you reacted. Setbacks are great to talk about in interviews if you have some intelligent analysis to go along with it."

Completing a project postmortem will help identify the issues that plagued your project (like the IGNEOUS team did on Pg. 32), but your "intelligent analysis" of those events is another crucial part of the process. In this article, we'll examine some common failures of student games and how to reflect on those challenges in a forward-looking way.

SCOPES GONE WILD

>> Everyone knows about scope issues, but narrowing down your game's focus is always easier said than done. It's so easy to overestimate the amount of work that can be done in a short amount of time that it's almost a given that your first few games will be overly ambitious. The time required for testing, tuning, and iteration is surprisingly long and easy to overlook during planning, especially when you're trying to cram in as many features as possible. If you don't plan correctly, you'll likely wind up with an unfinished game, as many students do.

"On my first student game, we all had very little idea of what we were doing," recalls Brenton Woodrow, a recent graduate of the Game Design program at Champlain College. "We completely over-scoped, dove in headfirst, and fell flat on our faces."

It would be easy to say that the lesson here is to be less ambitious at the start, or to multiply every work estimate by some factor to compensate for the time incurred by iteration and debugging. However, while more realistic time estimates can help, they're only one part of the equation. It's a fact of game development that no matter how padded the schedule, things will still go awry at some point.

Because schedule and scoping issues are a more a matter of "when" and not "if," the real lesson is learning how to prioritize and cut features. We'll discuss the art of prioritization in the next section, but first—the dreaded cut.

Learning to let go can be difficult, especially in the early stages of a game development career. You're in love with your idea or the geometry you built and the thought of it not making it into the game is difficult to entertain. But getting too attached to specific pieces of work can often work at cross-purposes to the overall quality of the game. It takes perspective to step back and to see that. Doing so will demonstrate that you're a developer with an experienced, mature view of how games are made.

TECHNICAL NIGHTMARES

>> Another problem that bedevils a lot of student projects is technology that proves to be a mismatch with the team, assets, or both. We've all seen projects where piles of beautiful art sat around in Photoshop or Maya waiting to be implemented until the very last moment—when it suddenly came to light that those assets wouldn't work as built because they were mismatched to the engine.

This is where we get to task ordering and prioritization, a crucial body of knowledge especially for those studying programming or production. Daniel Young, Adjunct Professor at Richland College (and a designer on DOOM 4 at id software) notes, "There's a huge temptation to do the 'fun stuff' first, at the expense of answering those important technical questions."

Playing around with higher-order systems like combat mechanics or player upgrades can be great fun because the rewards are so plainly in view. But working on those at the expense of determining how you're going to load those systems in the first place can—and often will—lead to trouble. Learning this means getting used to the fact that the first parts of many game development projects are often lacking in impressive looking, demo-friendly features.

No matter how short the project development cycle actually is, there's just no getting around the initial stage of work that proves how feasible your technology is in terms of getting the game working. The sooner this gets done, the faster you can get to the fun stuff.

Sometimes that means setting your original sights a little lower in terms of technical aspirations and deciding to focus on aspects that often get short shrift in student games, like balance and polish. "On one of my later projects, we chose a very simple goal: to make an arena combat game with one level and four characters, using a preexisting engine," says Woodrow. "Some of the other students criticized us for having low ambitions. However, that choice meant we could spend a lot of our time playtesting characters and weapons, tuning the actual gameplay. It ended up being one of the best, most fun games we ever produced."

Woodrow's arena combat game was very simple, using an existing engine and just four characters, which left a lot of time and room for tuning and balance.

THE BAD EGG

>> Sometimes a person's work just doesn't cut it. A programmer may claim his code was just a couple lines away from being fully functional, when in fact it was a broken mess. An artist might be the quiet type—so quiet that he doesn't speak up and say he thought the amount of art he agreed to do was actually impossible. Or one of the students on your team might have decided to switch into criminal justice or culinary school after this class, and can't be bothered with this game, particularly.

While it's frustrating to work with these people, it's important to resist the urge to lay the blame on others—especially when describing the problem in an interview later. Remember that the vast majority of game development is a team effort and that these kinds of problems are difficult to avoid any time you have a large group of people. Being able to describe the way the project failed while avoiding personal attacks or recriminations will not only show your ability to professionally deal with the kinds of challenging situations faced by game developers every day, but also demonstrate that you're thinking ahead, not stuck in the past.

Instead of framing your experience as a lesson to "not trust what your programmer says," think of it as an education in the necessity of possessing a flexible backup plan. Even the best designers, programmers, and artists can make mistakes and may fail to come through

for a variety of reasons. Having options to fall back upon helps ensure your project's success even in the face of unpredictable circumstances. Producers call this "mitigating risk," and it's something you can start doing just as soon as you allow yourself the freedom to change course if you encounter headwinds in your original direction.

PRISMATIC VISIONS

» "During one postmortem session of a student project that I watched," says Woodrow, "everyone seemed to understand that the game was intended to be a slow-paced tactical shooter. Suddenly one guy looked surprised—'Really?' he said. 'I thought we were making an action combat game.' The teacher just facepalmed at that moment."

Making sure that every team member understands the game's vision can be a surprisingly tricky thing. Just when you think everyone's agreed that the game is meant to be a certain way, people tear off in their own directions and only reconcile their goals after it's too late. (This effect can be compounded in school because students sometimes feel they have little incentive to follow the project direction, especially if that direction is weak to begin with.)

In order to make sure everyone has the same idea of what "success" for the project means, communication has to happen at all times, not just at the beginning. It's tempting to want to spend the bulk of your available time cranking out assets and writing code, especially when trying to put something together on short notice. But the limited development time actually means constant communication is all the more important because there's so little room for error.

But of course you do also need to have that clear vision at the start of the project. A good vision isn't about determining everything down to the letter in advance. In fact, at first there may be few answers as to how the vision will be achieved. But if you're comparing every step of the way to coherently articulated goalposts, each individual contributor's work will find itself gravitating toward a central locus, and your game will benefit from that coordinated effort.

DISCIPLINE BLINDERS

» A video game is a combination of many different elements, and only works as a game when all those pieces are in concert with each other. Neglecting any one of those aspects can cause problems for the game as a whole. "Sometimes, my students will start building levels without thinking about what the gameplay is going to be inside those spaces," Young says. "Or, they'll spend time coming up with a character, a story, and a world, but don't get around to developing anything that addresses the game mechanics."

If you've been on a project where an area of the game you weren't involved with held back the result, educating yourself on the jobs of your other team members will help you understand what went wrong, and also teach you some of the warning signs that may indicate a problem the next time around.

In today's era of large teams, specialization often presents itself as a viable option. But specializing too early can potentially limit your career. To take a longer view, the move to larger teams makes coordination

at the interdisciplinary level more vital than ever before. Because of that, an understanding of all aspects of game development can only help you. Don't pass up the opportunity to learn about them while you're still in school.

At some studios, you may be encouraged to believe that what happens in other departments is not your concern. But games are not just a few systems or pieces of art in a vacuum. If you're modeling a character, you have to think about how that character will be used in the game. If you're designing a weapon, it's absolutely necessary to consider the design's impact on art and engineering. Being able to interface with all disciplines is key, and if you can bring up your concerns in a respectful, polite way, you'll help ensure the success of your project.

ONWARD, TO VICTORY!

» A lot can be gleaned from postmortems, and those you read in *Game Developer* or Gamasutra can give you an idea of how to perform your own. While it's important to remember to not point fingers too much, you should be honest in your assessment of what went right and wrong with your project, and truly learn from it. Many developers keep making the same mistakes over and over, due to the simple fact that game development is a very complex process with a lot of moving parts.

This article describes just a few of the ways that game projects run into trouble, of course. But hopefully they've given you some food for thought in terms of determining what happened, and also how to interpret and discuss those results to people who ask you about them. The courage to describe your failed project honestly, combined with the curiosity to dig to the bottom of why it didn't turn out the way it was supposed to be will serve you well in your interviews and far beyond.

MATTHEW BURNS *is the founder of Shadegrown Games and formerly a producer at Bungie where he worked on the* HALO *series. E-mail him at* ***mburns@shadegrowngames.com****.*

PRIVATE COLLEGE

DigiPen Institute of Technology

Education, Location, Imagination

Since 1989 DigiPen has been helping students prepare for careers in the video game and animation industry with a formula of world-class faculty, an exemplary education, project-based curriculum and strong industry connections.

Top Five Reasons to Choose DigiPen Institute of Technology

// RECOGNITION

In 2010, The Princeton Review recognized DigiPen as one of the top two schools for game design in the United States and Canada. Electronic Gaming Monthly echoed this sentiment by naming DigiPen's program as the top game design school in the world. Aside from these publications *The Wall Street Journal*, *Rolling Stone*, *People* magazine, *USA Today* and *The Seattle Times* have also acknowledged DigiPen for their consummate and unique approach to education. Aside from the press, DigiPen students have been recognized 29 times at the Independent Game Festival (IGF) – more than any other school in the world – for their innovation in game programming. And, DigiPen Sophomore's secured the first win at the 2010 inaugural Indie Game Challenge.

// LOCATION

With its original campus in Redmond, WA, DigiPen Institute Of Technology is neighbored by approximately 150 game companies. As a result of DigiPen's proximity and proven success, DigiPen students have become the preferred candidates for many local and international organizations. With its established reputation and campuses in Singapore and Bilbao, Spain, DigiPen students and alumni have unparalleled access to potential networking and career opportunities.

// CONNECTED

DigiPen was North America's first game development degree program. For the past 20 years, DigiPen has garnered a strong reputation and has become known as "the place to go" if someone wants to break into the industry. Currently, Microsoft, Nintendo, ArenaNet, Valve, Bungie, Sony, Monolith, Big Fish Games, LucasArts, Rhythm & Hues, Intel, Activision, and many others have DigiPen alumni working within their organization.

// TEAMWORK

A DigiPen education is highly collaborative. Similar to the industry, to succeed as an individual you must succeed as a team first. Every year at DigiPen, students must form teams for a project class and apply their knowledge. Whether they are developing a game for IGF or an animated short film, students must create everything from "scratch". Which means students develop everything from complete game engines, to unique artificial intelligence, to story boarding, to sound development. This approach trains students how to think outside the box and be innovators, rather than users of software.

// SPECIALIZED

DigiPen is committed to providing an exemplary and focused education. Students at DigiPen spend a full four years in the their major, allowing them a greater depth of education over many other colleges. Also, DigiPen faculty are 92% full-time and are comprised of PhDs and experienced industry leaders. As a result, undergraduate interns from DigiPen have worked on published AAA title games, assisted in Microsoft research projects and have assisted in the development of the Wii controller.

Come see first-hand what sets DigiPen a part in higher education and in the industry by shadowing a student, sitting in on a class and/or attending an information session in person or online.

DIGIPEN OFFERS THE FOLLOWING DEGREE PROGRAMS

- Bachelor of Science in Real-Time Interactive Simulation
- Bachelor of Science in Computer Engineering
- Bachelor of Science in Game Design
- Bachelor of Arts in Game Design
- Bachelor of Fine Arts in Production Animation
- Master of Science in Computer Science (Part-time available)

"Because of DigiPen's focus on game design/development and real-world approach to the game development cycle, graduating students have a clear leg up on other college students seeking jobs in the games industry."

– James Pfeiffer, Test Manager
Microsoft Game Studios

DigiPen
INSTITUTE OF TECHNOLOGY

CONTACT INFO

DigiPen Institute of Technology
9931 Willows Road
Redmond, WA 98052
866.478.5236
admissions@digipen.edu
www.digipen.edu

The Los Angeles Film School

When you are ready to do more than simply play your favorite games, the Game Production Degree Program at the Los Angeles Film School in Hollywood, CA could be your next move.

The 18-month Associate of Science Game Production Degree Program is designed by professionals in the entertainment software industry to be a comprehensive and unique educational experience with an objective to provide a complete academic experience and produce graduates with an exceptional level of skill and competence.

During the program, you will learn to analyze and appreciate games from a rich multi-cultural and historic perspective. You will be challenged to create and showcase engaging game experiences using fundamental game mechanics while surveying current and emerging employment and money-making opportunities in the video game industry.

Creating original 2D and 3D art and adeptly manipulating these assets will allow you to develop your artistic technique. Students also advance their coding skills with scripting languages by learning programming basics through writing mini-games.

Creative and technical considerations are explored using a digital platform in design courses as you experiment and expand your design skills. Prior to your Final Project, you will have the opportunity to conceive, refine and pitch an original game concept to a "greenlight committee."

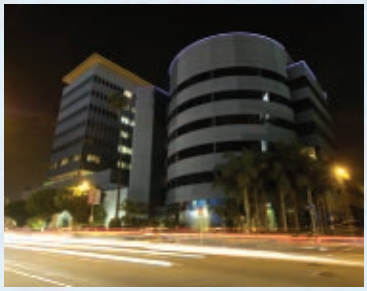
During the greenlight process, you will collaborate with and compete against other students for the chance to produce your own original game project. Once the Final Projects are greenlit, you will begin the pre-production process – constructing the game design document and technical design document – as you make your final planning arrangements before entering the production stage. The Final Project will demonstrate a culmination of skill, teamwork, planning, technique, final presentation and professionalism in producing a playable game or map.

Spend quality time with our faculty, get up close and personal with industry guests and log real time in a simulated work environment as you prepare for the gaming career of your dreams. Because a foundation in general education is as important as knowing the most current techniques and equipment, you will also learn basic industry skills that include time management, self-motivation, resume writing, communication skills and how to prepare for that first step into the industry. And with your success in mind, The Los Angeles Film School's Career Development Department is there to provide guidance as you interview for internships and entry-level jobs in the entertainment media industry. In addition, our career development services continue to be available to you throughout your career.

Classes start bimonthly; apply all year long. In addition, all of our enrolled students receive a gaming laptop loaded with software needed to enhance your learning experience (additional required fees apply).

// FIND OUT MORE

Visit lafilm.edu or call 877-9LA-FILM.



"The Los Angeles Film School's state-of-the art facilities are a veritable treasure chest for aspiring artists and creatives."

– Terrence Myers, Director
Membership & Awards Production
Academy of Interactive Arts & Sciences

"Entertainment interchanges from movies to games all of the time. Los Angeles is an ideal place to learn because it presents a great opportunity for those involved in graphics and movies."

– Nolan Bushnell, Founder of Atari
Member of LAFS Game Production
Advisory Committee

THE LOS ANGELES® FILM SCHOOL

FILM + GAMES + ANIMATION + AUDIO

CONTACT INFO

The Los Angeles Film School
6363 Sunset Blvd.
Hollywood, CA 90028
323.860.0789 or 877.9LA.FILM
info@lafilm.edu

www.lafilm.edu

Vancouver Film School

Game Design at Vancouver Film School is an intense one-year program that covers everything you need to join the game industry as a designer or producer, from theory to hands-on practice to the production of a professional-quality portfolio. There's a reason why the *L.A. Times* called VFS one of the top 10 schools "favored by video game industry recruiters."

VFS Game Design students learn more than just one side of game design - they experience the full scope of this varied and rewarding career through an in-depth curriculum that includes:

- » Interactive Narrative
- » Analog Games
- » Interface Design
- » Scripting
- » Level Design
- » Pre-Production
- » Project Management
- » Flash
- » Mobile & Handheld Design
- » Game Audio
- » The Business of Games

// LED BY INDUSTRY

In VFS Game Design, you're mentored by a faculty of respected industry pros – your first crucial connections to the professional world. At the helm is veteran Dave Warfield, who, as a Senior Producer for EA, helped produce and design the *NHL* franchise for 10 years. His many other credits include titles like EA's *NBA Live* and Konami's *Teenage Mutant Ninja Turtles*. An Advisory Board of industry leaders, including luminaries from Irrational Games (2K Boston), Microsoft, Nokia, and Ubisoft, keeps the curriculum on the cutting edge.

// A STUDIO ENVIRONMENT

In a process that closely mirrors a real-world studio environment and production pipeline, you work in teams to take games from concept to completion. Toward the end of your year at VFS, you get the chance to present your final playable games to an audience of industry representatives and recruiters: a unique chance to prove yourself and make valuable professional contacts.

// LIVING & CREATING IN VANCOUVER

In VFS Game Design, you have the advantage of learning in Vancouver, B.C., Canada. Along with strong its film, TV, and animation industries, Vancouver is a world center of game development, meaning that VFS is always industry-current, hosts many guest speakers, and provides you with vital mentorship and feedback opportunities throughout your year. It's the perfect place to get your career started.

// THE RESULTS

Our graduates have gone on to earn key design and production roles at top studios around the world. A small selection of their recent and upcoming titles includes: *Warhammer 40,000: Dawn of War II*, *Marvel Ultimate Alliance 2*, *Prototype*, *Dragon Age: Origins*, *Punch-Out!!*, *Mass Effect 2*, *FIFA 10*, *Skate 3*, *True Crime*, *Dead Space 2*, *Star Wars: The Old Republic*, *Dead Rising 2*, *Pirates of the Caribbean: Armada of the Damned*, and *ModNation Racers*.

Find out about VFS Game Design and begin your career at vfs.com/gamecareer.

"It was a really mind-opening experience for me, in terms of what is possible creatively. VFS was really instrumental in me being successful today."

– Armando Troisi, VFS Graduate
Lead Cinematic Designer
Mass Effect franchise

VFS VANCOUVER FILM SCHOOL
Results Matter

CONTACT INFO

Vancouver Film School
200-198 West Hastings St
Vancouver, BC V6B 1H2
Canada
604.685.5808 or 800.661.4104
inquiries@vfs.com

www.vfs.com/gamecareer

Masters of Digital Media Program @Great Northern Way Campus

CENTRE FOR **DIGITAL MEDIA**

MASTERS OF DIGITAL MEDIA PROGRAM | GREAT NORTHERN WAY CAMPUS

Our Academic Partners:

University of British Columbia | Simon Fraser University | emily carr university of art + design | British Columbia Institute of Technology

MASTER'S DEGREE IN DIGITAL MEDIA ANSWERING INDUSTRY'S CALL FOR TALENT

The Masters of Digital Media (MDM) Program, Canada's first professional graduate degree of its kind in digital media, is developing the next generation of leaders in digital animation, video games, interactive design, e-Learning and virtual world technologies.

The MDM Program offers a challenging academic curriculum relevant to the needs of the digital media industry. The team-based, award-winning program provides real world experience through access to industry partners such as Electronic Arts, Autodesk, Rainmaker, Radical Entertainment – Vivendi Games, Propaganda Games-BVG Disney and Microsoft. Further, students work on industry-funded projects and on four-month internships.

But it's not all work! Movie nights, multi-player *Rock Band* sessions and pizza Fridays take the edge off the high performance week.

// A DISTINCTIVE DEGREE FOR A DISTINCTLY OUTSTANDING STUDENT BODY

The MDM Program is housed at the Centre for Digital Media located at Great Northern Way Campus (GNWC). GNWC combines the strengths of four leading academic institutions: the University of British Columbia, Simon Fraser University, Emily Carr University of Art + Design and the British Columbia Institute of Technology. MDM graduates receive a powerhouse Master's degree bearing the seals of all four academic partners.

The MDM Program is targeted at individuals with an undergraduate degree in a variety of disciplines. MDM classrooms are populated with artists, computer scientists, film-makers, entrepreneurs, philosophers and engineers who all share a passion for digital media. Each class has a mixture of seasoned professionals who have worked in the industry and students continuing directly from an undergraduate program.

// LOCATION, LOCATION, LOCATION: VANCOUVER, CANADA – THE CENTRE FOR DIGITAL MEDIA

The MDM can boast not just a world-class program, but a world-class city to offer it in. Vancouver is home to more than 1,100 digital media companies and numerous industry leaders and innovators in digital film, television, video games and interactive advertising. There are more videogame developers per capita in Vancouver than any other city worldwide. The city is an excellent environment—unparalleled anywhere in the world—to nurture and teach our brain trust of digital media minds.

Situated on Canada's West Coast, Vancouver is the Gateway to Asia and one of only a few places where it is possible to snowboard, hike and sail - all on the same day. This dynamic and diverse city (BC's largest) is consistently ranked as one of the most livable on the planet. In 2008, Vancouver was named the "Most Liveable City" in the world by the Economist Intelligence Unit. Next year, it will host the 2010 Olympic and Paralympic Winter Games.

// BECOME A MASTER OF DIGITAL MEDIA

The 20-month MDM Program is cohort based and commences each year in January and September.

Apply now at **mdm.gnwc.ca**

"The MDM represents a once in a lifetime opportunity: To be part of something new and exciting that has no equal anywhere else in the world."

– Ashley Blacquiere, MDM Graduate
Class of 2009
+ Mission Designer
Radical Entertainment

"The MDM Program builds upon solid academic fundamentals, and mixes that with industry expertise and experience, preparing students for leadership positions in our industry upon graduation."

– Don Mattrick, Senior VP
Interactive Entertainment Business
Microsoft

CONTACT INFO

Masters of Digital Media Program
@Great Northern Way Campus
577 Great Northern Way
Vancouver, BC
Canada

Speak to an advisor:
Alison Robb
778.370.1031
alison_robb@gnwc.ca

mdm.gnwc.ca

MEDIA ARTS COLLEGE

Madison Media Institute

In fall 2009, Madison Media Institute launched its long anticipated Game Art and Animation program. Madison Media Institute (MMI) is a media arts specialty school celebrating over 40 years in the education field with campuses located in Madison, Wisconsin and Minneapolis, Minnesota.

The program was developed by industry veteran Eric Weiss, with over 20 years experience in computer animation production. Eric has worked on several best-selling game titles including: Electronic Arts' *James Bond, From Russia with Love*, *The Lord of the Rings, The Third Age*, and *Prey*, developed by Madison's Human Head Studios. Eric also has extensive experience in film animation and visual effects and his credits include *Superman Returns*, *The Polar Express*, *Final Fantasy*, *The Spirits Within* and *Godzilla*. Eric recently completed work as the Digital Supervisor on Nickelodeon's hit show *Back at the Barnyard*.

Leaders from local gaming companies in Madison were also instrumental in the development for course make up, content, software and textbook choices. Having gaming companies participate in the creation of the program is just the beginning of their involvement, as they will also have continued presence in the advisory committee, student portfolio reviews and guest lectures.

The Game Art and Animation program concentrates on the artistic side of the game production process and graduates of the program are prepared for entry-level jobs as modelers, animators and technical animators. Students have courses in all of these areas but also have the opportunity to focus their skills in a chosen discipline in both team-based and individual projects. Maya has been chosen as the 3D animation software program as it has become the industry standard animation package. By the end of the 75 credit hour program, graduates are highly proficient with this comprehensive tool. For modeling, students will learn to create 3D models using Maya, Mudbox and ZBrush as well as Body Paint and Painter. They work on human and non-human characters, environments and backgrounds as well as props. Animation courses will include Maya, Motion Builder and hands on experience with Motion Capture performances. Students also learn the technical animation skills of character rigging, programming with Maya's Embedded Programming Language (MEL), python scripting and advanced lighting, rendering and compositing techniques.

Throughout the program, students learn how to integrate their animation and models into a working game using the Unreal Game Engine by Epic Games. This is a cornerstone of their training, along with motion capture experience, as it allows their graduates to hit the ground running on their first jobs when they begin their careers. Learning how game art is integrated into a game engine is essential in the optimization of game art in a production environment. Students also graduate with a complete portfolio including a demo reel, DVD , website and working game level created with their own art. Graduates from the Game Art and Animation Program at the Madison Media Institute will stand out above the rest.

For complete details of the Game Art and Animation Program, please visit www.madisonmedia.edu or email admissions@madisonmedia.edu.

"Raven Software is thrilled to be a part of the Advisory Board of the Game Art and Animation Program at the Madison Media Institute. As one of the top game development companies in the Midwest, we are looking forward to continuing our relationship with MMI."

– Ravensoft is a Division of Activision and located in Madison, Wisconsin. Its 2009 releases include *Wolfenstein*, *X-Men Wolverine* and *Singularity*.

"Human Head Studios, located in Madison, Wisconsin, is excited to work with Madison Media Institute on it's new Game Art and Animation program and we are looking forward to assisting MMI on the program's Advisory Board."

– Human Head Studios releases include *Rune* and *Prey*.

MADISON MEDIA INSTITUTE
College of Media Arts

CONTACT INFO

Madison Media Institute
2702 Agriculture Drive
Madison WI 53718
800.236.4997
admissions@madisonmedia.edu
www.madisonmedia.edu

Centre for Distance Education

"This is the way I like my education."

— KEVIN GRAHAM, GRADUATE

CD-ED's Digital Arts Technology Training Institute offers a one-year fully recognized 3D Game Artist diploma program available in Canada, the United States, the United Kingdom and any other country. With a main campus in Sydney, Nova Scotia and a new branch campus in Tampa, Florida, plus the virtual campus that allows global access, it's never been easier for a student to obtain an education.

While developing the 3D Game Artist program, we took our cue from the gaming industry itself and created a program that prepares successful graduates for entry-level employment. The curriculum is based on requirements listed in actual employment ads from companies such as Bioware, Blizzard Entertainment and the Walt Disney Company.

With a detailed focus on precise skills, our students are ready for positions such as 3D Background Artist, 3D Artist or an Environment Artist. As a CD-ED graduate, you will have all the 3D techniques and skills required for a variety of positions. Imagine yourself creating low-poly and high-poly models based on assigned concept artwork and model sheets while adhering to the polygon and texture budgets established for the production. Or, you could use your new skills to find a position at a company creating games for the PC or for independent gaming platforms such as the PS3, DSi, Wii, PSP and more.

At the end of the 3D Game Artist program, our students can prepare models to accept custom game-specific textures with UV maps. They know the technical aspects of using normal mapping and texture baking to bring high-poly detail into a low-poly game setting and how to import 3D assets into a game engine.

CD-ED's learning methodology places a significant focus on portfolio development. As every hopeful game artist knows, the portfolio will make you or break you. Upon graduation, our students have both a CD-ready portfolio and a web portfolio.

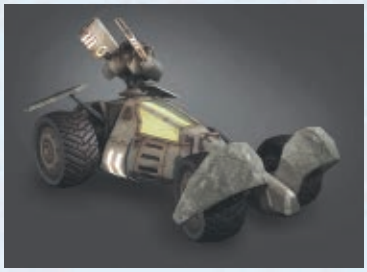
This innovative, industry-driven, fully online and interactive program immerses our students in the digital world to the degree necessary to truly excel in the gaming industry. Our graduates are in high demand. CD-ED graduates are compositing and modeling at IMAX and contributing to XBOX 360 games.

CD-ED has the courses employers want, when and where the student wants to take them. Online education means better choices. With minimal disruption to your daily life, you can become a 3D Game Artist regardless of where you live or work.

After graduation, we will help you find an interview with your prospective employer in the job market and coach you through the job hunt until you find your dream job.

CD-ED is committed to helping our students find employment. In fact, we'll start the job search before you even begin as a student. If you call us today, your personal admissions representative will research the job market for this career in your city, and help you decide if the 3D Game Artist program is right for you.

Apply now to ensure you are accepted on your own schedule.



"I got recruited as a 3D Artist with Hydrant Studios Inc. in the Niagara region. We work on XBOX 360 titles developing assets and characters. As a 3D artist at Hydrant, I am responsible for concept development (character and environment assets), 3D modeling, texturing, sculpting and preparing the assets to be rigged and animated. My tasks are always different from project to project. One day I can be working on a vehicle or a building and the next day I can be making a three-legged creature from a fantasy swamp."

– Alejandro Mendoza, Graduate

CENTRE FOR Distance Education

CONTACT INFO

Main Campus: Canada Division
Centre for Distance Education
222 George St., Suite C
Sydney, Nova Scotia
Canada
B1P 1J3

Branch Campus: USA Division
Centre for Distance Education
14502 North Dale Mabry Highway
Suite 200
Tampa, FL 33618

866.446.5898 or 866.567.3010
info@cd-ed.com

www.cd-ed.com

Florida Interactive Entertainment Academy

FIEA games have won awards from MTV, EA, IGF and Shockwave.

Students learn how to rapid prototype in the first semester.

Every student gets their own laptop and workspace.

Part of the University of Central Florida, the Florida Interactive Entertainment Academy (FIEA) offers a graduate gaming education in a world-class facility in downtown Orlando. It teaches artists, programmers, and producers the techniques, tools and skills that you don't get in a traditional degree program.

You'll be exposed to an industry-based curriculum that mimics the production environments of successful gaming studios. You'll work on student teams with real-world projects to learn the skills and software of video-game design and development. In just 16 months, you'll earn a Master's of Interactive Entertainment.

"Attending FIEA was the smartest decision I ever made. There's no doubt that I wouldn't be where I am today without going there."

– Matt Read, Lead Designer/Sr. Designer
Electronic Arts
Madden NFL 8, 9, 10, 11
Madden NFL Arcade

Five Reasons to Choose the Florida Interactive Entertainment Academy

// FACULTY

Our faculty are industry veterans with real-life answers. Collectively, FIEA faculty have shipped more than 40 of today's biggest games and films and have worked in studios like EA, Disney, Microsoft and Take Two. And our 8-to-1 student-to-faculty ratio ensures you'll get the attention you need.

// MASTER'S DEGREE

Unlike some schools, FIEA offers a fully accredited Master's of Interactive Entertainment. So you not only get a cutting-edge education but also a degree that's recognized and a valuable marketplace commodity. And you get it in only 16 months.

// INDUSTRY SUCCESS

Bioware. Zynga. Neversoft. Cartoon Network. Irrational. Bethesda. Electronic Arts. FIEA grads are everywhere making games and building their careers.

Recent games FIEA grads have worked on include *Madden NFL Football 10*, *Call of Duty 4*, *Guitar Hero World Tour*, *Tiger Woods 10*, *Rock Band 2* and *Marvel Ultimate Alliance*.

// THREE TRACKS

Our producer, programmer and artist tracks teach you all you need to know to become industry ready. As you specialize, you'll go deep into your chosen discipline to learn all the techniques and tools used to make AAA games and films. But because we never offer two classes at the same time, you can also "minor" in one of the other specialties to learn valuable additional skills.

You'll apply these skills in our team-based curriculum. From the day you enter, you'll be put on a game team with other artists, programmers and producers and be required to display the communication and commitment necessary to bring a game to fruition.

// THE SPACE

FIEA is located in UCF's Center For Emerging Media where FIEA students collaborate with student musicians, filmmakers, architects and animators on games and other interactive projects. The 114,000 sq. ft. building features a soundstage and motion capture facility that host student and industry shoots.

Learn more at www.fiea.ucf.edu

FIEA Florida Interactive Entertainment Academy

UCF Center for Emerging Media

CONTACT INFO

Florida Interactive Entertainment Academy
University of Central Florida
Center for Emerging Media
500 West Livingston St.
Orlando, FL 32801
407.823.2121
info@fiea.ucf.edu

facebook.com/UCFFIEA
twitter.com/FIEA

www.fiea.ucf.edu

The Art Institutes

TRANSFORM YOUR TOMORROW

Game Art & Design at The Art Institutes

You may think you're just sitting in a room by yourself playing video games. But the fact is, you're setting yourself up to join a growing movement of creative thinkers. People who use their individual style and imagination to energize the economy, make an impact on their industry, and create their own tomorrow.

Your creativity has you ready to plug into this new creative economy and seize the opportunities it offers. The key is getting the right education at a school that gets how creative people think and knows how to help you unleash your talents so you can make the leap from game player to game developer.

Not just a school. A launching pad for your future. It starts with your creativity. Comes to life in the form of a character. And moves, breathes, and exists through the tools, the technology, and the educational experience at an Art Institutes school.

At an Art Institutes school that offers Game Art & Design programs, you'll find yourself in a creative community. Learning hands-on alongside other students who share your passion for gaming. Working with instructors who work in the same areas they teach, so they can keep up on industry trends, understand the demands of the career you're preparing for, and know the tools and technologies you'll use in your job.

They're mentors who'll guide you, encourage you, and prepare you to start making an impact on the world of gaming. And they're doing a great job; The Princeton Review's latest edition of the "Top 50 Undergraduate Game Design Programs" ranks The Art Institute of Vancouver at #6.

An education that can prepare you for tomorrow. In our gaming programs, you'll add the textures, devise the challenges, and set the parameters that make your games uniquely yours. Our fundamental courses will get you started, and then you'll move on to the real tools of the trade—drawing, color, design, computer applications—you'll even study anatomy so you can create more lifelike characters. Then it's on to image manipulation, cinematography, creative storytelling, storyboarding, and 2D and 3D modeling techniques, using industry-related equipment.

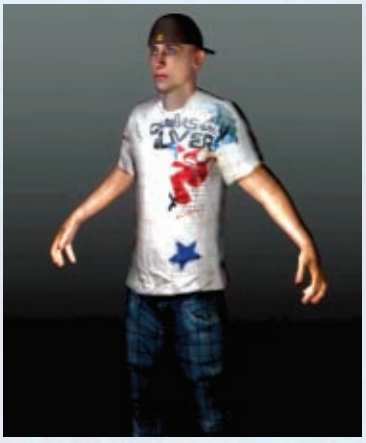
Get ready to compete. Gaming is a competitive industry. So you'll build a portfolio that'll show employers what you can do. We even offer Portfolio Shows where you can share your work with companies who hire graduates from Art Institutes schools because they know they can count on our grads to deliver.

After graduation you can qualify for an entry-level job as a game tester/analyst, game designer, level designer, texture artist, cinematic artist, 2D artist, or 3D artist in a software, game design, or education company.

Get started creating your tomorrow. Take the next step. Contact us and we'll help you get started toward creating a rewarding tomorrow in Game Art & Design.

The Art Institutes is a system of over 45 schools throughout North America. Financial aid is available to those who qualify. Program offerings vary by school.

Combine your creative talent with the right technology, and you can bring every image, character, or setting to life—from the big picture down to the smallest detail.
Student Shown: Yusuke Sato, Game Art & Design
The Art Institute of Atlanta, 2009 Graduate



Building your digital self? It's more than just wrangling pixels. It's focusing all your talent towards a real future in gaming.
Student Artwork: Brandon Kozarek, Game Art & Design
The Illinois Institute of Art—Chicago, 2007 Graduate

Student Artwork:
Jacob Ligon,
Game Design &
Programming
The Illinois Institute
of Art—Schaumburg,
2009 Graduate

The Art Institutes™
America's Leader in Creative Education

CONTACT INFO

The Art Institutes
Administrative Office
210 Sixth Avenue, 33rd floor
Pittsburgh, PA 15222
800.894.5793

gamer.aii.edu

Savannah College of Art and Design

The Savannah College of Art and Design offers Bachelor of Arts, Bachelor of Fine Arts, Master of Arts and Master of Fine Arts degrees in interactive design and game development. Students can earn their degrees at the university's Savannah and Atlanta, Georgia, locations and online via SCAD eLearning. Additionally, SCAD now offers U.S. bachelor's and master's degrees in Hong Kong.

No matter the location, the interactive design and game development program at SCAD fuses artistic and technical training to prepare students for professional, creative careers in the rapidly growing interactive and game industries. Students develop the aesthetic, artistic, design and technical abilities necessary to become leading artists, level designers, game designers, innovators, and interactive Web and experience designers, as well as installation and digital artists.

// ENROLL IN A TOP-RANKED PROGRAM

According to the Los Angeles Times, the SCAD interactive design and game development program ranks among the top 10 in the United States—out of more than 200—mentioned by industry recruiters.

// USE HIGH-END HARDWARE AND SOFTWARE

At SCAD, students use a motion-capture lab, green screens, Cintiq monitors, Autodesk Maya, Z Brush, Photoshop, Illustrator, Premiere, After Effects, Flash, C++, Unreal game engine, ActionScript and Dreamweaver.

// LEARN FROM TALENTED, DEDICATED FACULTY MEMBERS

SCAD's accomplished faculty of artists, designers and programmers includes award winners and experts in their fields. They bring to the classroom a combined total of more than 50 years of experience in both game development and interactive design.

// NETWORK WITH INDUSTRY PROFESSIONALS

Representatives from leading game design and digital media companies—including Lucas Arts, Activision, Blizzard and Electronic Arts—frequently visit SCAD to recruit students and new alumni.

// PARTICIPATE IN CONFERENCES

SCAD sponsors the annual Game Developers eXchange, which brings together game developers, educators and students with experts who share their behind-the-scenes knowledge. Faculty and students also regularly participate in the IGDA Leadership Forum, the Game Developers Conference and SIGGRAPH.

// JOIN OTHER AWARD-WINNING STUDENTS

In March, undergraduate interactive design and game development students Gwen Murray and Seth Smith won the ScreenBurn at SXSW Game Design Competition at the South by Southwest Interactive Festival.

Follow in the footsteps of alumni employed by top game design companies.

SCAD Alumni are currently working in the most prestigious interactive and game production studios, including Blizzard, Activision, Electronic Arts, Lucas Arts, Kaos, Firaxis, Zynga, and RG/A.

// FIND OUT MORE

Visit www.scad.edu/interactivedesign, call 800.869.7223 or e-mail info@scad.edu.

// SCAD: THE UNIVERSITY FOR CREATIVE CAREERS

The Savannah College of Art and Design is a private, nonprofit, accredited institution that offers bachelor's and master's degree programs in 42 majors. Visit www.scad.edu.

"The wide variety of skills I have gained at SCAD continuously proves to be extremely valuable. My experience from SCAD has given me the confidence and ability to excel in various aspects of the development process at Zynga. I will always be thankful for the knowledge and inspiration my professors passed on to me."

– Thomas Kastner
Game Designer at Zynga

I've have recently had the pleasure of participating in a five year curriculum review for the Interactive Design and Game Development department at SCAD. This review revealed the high level of quality of the programs, facilities, faculty, and students that make up the school. "

– Luis Cataldi, Art Director
Kaos Studios/THQ

SCAD
The University for Creative Careers®

CONTACT INFO

Savannah College of Art and Design
P.O. Box 2072
Savannah, GA 31402-2072
912.525.5100 or 800.869.7223
admission@scad.edu

www.scad.edu

Full Sail University

World-Class Game Programs at Full Sail University

Delivering an educational experience that mirrors the workflow and technology found at today's major gaming studios, Full Sail University offers multiple degree programs targeted toward specific roles in the game creation process. Located in Winter Park, Florida, the school's programs are developed for the diverse career opportunities found in the video game industry, whether a student is interested in programming and development, art and design, or production and project management.

// GAME ART BACHELOR'S DEGREE PROGRAM

Delivered both on campus and through the school's innovative online platform, the Game Art Bachelor of Science Degree Program presents a heavy emphasis on character development, shading and lighting, texturing, and modeling – giving students a solid foundation in the art production methods and workflow seen in the industry. Game Art students also collaborate with programmers and designers from other Full Sail programs to create art for a playable game project.

// GAME DEVELOPMENT BACHELOR'S DEGREE PROGRAM

Students who are interested in writing the code behind today's most innovative games can thrive in Full Sail's on-campus Game Development Bachelor of Science Degree Program. A comprehensive programming degree, it teaches students the latest processes for creating single and multi-player games for networks, consoles, handheld devices, and computers. The courses mimic studio environments, where students learn essential skills in artificial intelligence, C++ programming, 3D graphics, and story development, while creating their own playable games.

Full Sail graduates have worked on the following projects and more:

- *Call of Duty: Modern Warfare 2*
- *Diablo III*
- *Dead Space*
- *Madden 2011*
- *Marvel Ultimate Alliance 2*
- *Midnight Club: Los Angeles*
- *Singularity*
- *Red Dead Redemption*

// GAME DESIGN BACHELOR'S DEGREE PROGRAM

Behind every gaming experience is a story – and a team of professionals that brought it to life. Offered exclusively as an online program, the Game Design Bachelor's Degree Program is designed for students who want to create compelling games, with coursework that focuses on game play, story development, and the production needs of the industry.

// GAME DESIGN MASTER'S DEGREE PROGRAM

Courses in the Game Design Master of Science Degree Program stress the creative production and business management skills used at professional game studios. Students learn to broaden their abilities as project coordinators, with practical applications that recreate the processes used by successful managers – including developing an understanding of team leadership, project development, and collaborative design.

// ACCOMPLISHED ALUMNI, INDUSTRY ACCOLADES

The range of skills covered in Full Sail's degree programs has enabled graduates to go on to successful careers with major game studios across the globe, including Electronic Arts, EA Tiburon, LucasArts, Epic Games, Rockstar Games, and n-Space.

Full Sail University's game degree programs have also received a number of honors within the gaming industry and beyond, such as being named "The Harvard of Game Schools" by *Tips & Tricks* magazine, and one of the Top Five Game Degree Programs in the World by *Electronic Gaming Monthly*.

Named one of the Top Five Game Degree Programs in the World by *Electronic Gaming Monthly*

"The Harvard of Game Schools"

– *Tips and Tricks* magazine

FULL SAIL UNIVERSITY®

CONTACT INFO

Full Sail University
3300 University Boulevard
Winter Park, FL 32792
800.266.7625 or 888.993.7338

www.fullsail.edu

Westwood College

Video games. You don't just love to play them. You love and appreciate everything that goes into making them. It's an interactive world where design, software development and animation intersect. And, you want in. But have you ever wondered how you can have a role in actually creating them? If you're looking to turn your passion for gaming into something bigger than just a high score, then you're ready for Westwood College.

Westwood has two different paths to achieve a career in the game and interactive industry. The first is through our School of Technology, where you'll learn what it takes to develop and program interactive games with a degree in Game Software Development. Or pursue a gaming career through our School of Design where you'll learn to create, design and animate with either a degree in Game Art or Animation. Each program gives you a jumpstart to your career and the foundation that will help you succeed.

// GAME SOFTWARE DEVELOPMENT

A bachelor's degree in game software development from our School of Technology gives you the core knowledge of software engineering that employers are looking for. Our coursework will teach you how to apply critical thinking, logic, communications and problem-solving skills to the game development environment. And with your degree, you'll be ready for a career as an application developer, game software programmer, product software developer, game designer or game tester. Our course topics include:

- **Team Product Development**
- **Fundamentals of Game Engine Development**
- **Artificial Intelligence for Games**

Upon completion of our game software development program, you will be able to design, develop, test and deploy a game or interactive software product. Our graduates are working for some of the industry's biggest names, including*:

- **Activision**
- **EA Games**
- **NetDevil**

// GAME ART

In our game art program within the School of Design, you'll develop key knowledge of art and animation techniques like traditional 2D artistry and 3D modeling, while learning how to analyze, troubleshoot and solve complex programming assignments. When you graduate, you'll be ready for positions such as a 3D character modeler, lead artist, or quality assurance game tester. Our course topics include:

- **Texture and Mapping for Games**
- **Intermediate 3D**
- **Advanced Drawing and Perspective**

Upon completion of our game art program, you could follow the path of past graduates, and you too could be working at places like*:

- **Blizzard Entertainment**
- **Sony Online Entertainment**
- **Vivendi Games**

// ANIMATION

A career in animation is about bringing ideas to life. Through Westwood's School of Design, you'll develop core competencies like web page design and digital effects, while gaining the necessary fundamentals like how to draw and the laws of motion. As part of a well-rounded education, business courses teach concise communication skills and the ability to reason and problem-solve. Each graduate enters the workplace with the expertise they need to excel in today's marketplace. Course topics include:

- **Character Development and Design**
- **3D Modeling**
- **2D Animation**

Upon completion of our animation program, you'll be prepared to pursue positions like web animator, 3D animator, desktop publisher or graphic designer.

Westwood College is more than a place to get your degree. It's a place where you can jumpstart your career in as little as three years. Our unique accelerated approach helps you enter the workplace sooner, armed with the tools you need to compete in today's job market. Enroll in any one of our fast-track programs either online or at one of our campuses and discover that Westwood is right for you. With Westwood's outstanding program that will set its graduates apart from the rest, it's time to put your passion for video games to work.

** Westwood College must track the initial employers of its graduates for regulatory and accreditation purposes. This list is taken from a compilation of employers of Westwood College graduates from each region from June 2007 to June 2008. This is not intended to represent an endorsement of Westwood College by the listed employers.*

"I am exactly where I always wanted to be. I'm a concept artist at an independent video game developer working on the biggest video game titles in the world. Over the last two years I've had the opportunity to work on properties such as Halo and Call of Duty. The thing I enjoy most about my job is feeling creative. Every day I get the opportunity to make up things that don't exist yet. It's an immense challenge but extremely fulfilling."

– Jason Borne, Westwood Graduate

WESTWOOD COLLEGE | Game Software Development / Game Art / Animation

CONTACT INFO

www.westwood.edu/gaming

CAMPUS LOCATIONS

Denver // Atlanta // Los Angeles // Chicago // Arlington // Annandale // Online

triOS College

When triOS College set out to create a Video Game program, we turned to the Video Game Industry and posed the question: What skills are you looking for when hiring people for your company?

The results were clear. The Video Game Industry needs Video Game programmers with an understanding of design but solid training in areas including 2D Game Programming, Video Game MODification, Video Game Data Structures , Mobile Game Programming, 3D Game Programming and XNA Game Development.

According to industry experts, the career opportunities for game developers outnumber game artists 10:1. If you would like to start a career in video game development/ programming, making good money with great career opportunities, then we have the program for you.

Our program combines video game development training with intensive gaming projects and a real world 16-week industry internship at a video game company.

Graduates of this program will enter the workforce with 5 video game projects

in their portfolio, including Video Game Prototype, 2D PC Video Game, Video Game MOD, Cell Phone Video Game and XBOX Live Arcade Video Game (using Microsoft XNA).

Areas of training include: Video Game Core Foundations: Video Game Analysis and Technical Design, Video Game Business, Video Game Mathematics, Video Game Physics, Video Game Core Prototyping: Video Game Prototyping with Project, Intro to Programming with C, C++ Programming, MODification Video Game Development: Video Game Animation for Programmers, Video Game Level Design, Video Game MODification Project, Mobile / Online Video Game Development: Video Game Data Structures and AI, Java Programming, Mobile Game Development with Project, XNA Video Game Development: 3D Video Game Programming, XNA Video Game Development, XNA Video Game Project, 2D Video Game Development: Video Game Art and Modeling, 2D Video Game Programming, 2D Video Game Project.

Apply early as there are a limited number of spaces available for each start date. Students will receive a state-of-the-art gaming laptop during the program, combined with hands-on training, actual gaming projects and an Internship aimed at making you job-ready when you graduate.

"triOS College is planning on graduating the type of programmers that the games industry needs; I want to hire them now!"

– Keith Makse, CEO
Cerebral Vortex Games

"I wish I could take this program!"

– Tim Maly
Independent Video Game Producer

"Every course offered by triOS College has been designed by a compilation of industry veterans. This ensures that every graduate has a full knowledge of what is currently utilized in the industry."

– Hamed Abbasi
Vast Studios Inc.

triOS COLLEGE
BUSINESS ART NOLOGY HEALTHCARE

CONTACT INFO

triOS College
425 Bloor St. E., Suite 200
Toronto, Ontario
M4W 3R4
Canada
888.805.0533
info@trios.com

www.getintothegame.ca

Drexel University RePlay

Drexel University's RePlay is ranked in the Top 3 for Undergraduate Game Design in North America by the *Princeton Review* and *GamePro* magazine. Drexel RePlay is a collaborative effort between the Digital Media Program (in the Westphal College of Media Arts & Design) and the Computer Science Department (in the College of Engineering). At Drexel University, game development does not "live" in solely one department, and so mirrors the true nature of game development in commercial settings. Digital game development is offered in a coordinated, cross-listed series of courses in both the Computer Science (CS) and Digital Media (DIGM) majors, and production courses are open to other majors as well. Drexel offers both undergraduate (BS) and graduate degrees (MS in DIGM, MS and Ph.D in CS) in each program.

» Drexel's Digital Media Program offers a Game Art and Production major that instructs students on the foundation skills of design, art, programming, 3D modeling, animation, audio, and video production, and the use of industry standard tools.
www.digm.drexel.edu

» Drexel's Computer Science Department offers a Game Programming and Development concentration that instructs on foundation software development skills, and offer software design courses for prototyping game concepts.
www.cs.drexel.edu

» The RePlay gaming courses and projects bring these two majors together, with the additional participation of students and faculty from other majors including Music, Music Industry, Screenwriting and Playwriting, Engineering, Business, etc. The Gaming Workshop sequence features group projects involving dozens of team members.

» Our facilities include more than 120 workstations, a 16 camera Vicon motion capture studio, green screen, FTIR multitouch displays, laser scanner, stereoscopic projector, eye tracker, fNIR and EEG brain interfaces, and recording studios. Students use professional software including Unreal, Unity3D, Maya, 3D Studio Max, Houdini, and Massive.

» Drexel alumni include Jack Wall, award-winning composer for *Myst III:Exile* and *Myst IV:Revelation*, and Tom Fulp, creator of *Alien Hominid* and founder of Flash gaming portal Newgrounds.com. Recent graduates have received jobs from Microsoft Game Studios, Walt Disney Imagineering, Midway Games, NCSoft, Dreamworks, and Pixar.

» Games produced by Drexel students have won numerous awards, including the prestigious Adobe Achievement Award, the SIGGRAPH Game Jam, and the Philadelphia Game Jam. Drexel RePlay has been featured in *Wired* magazine, the *LA Times*, the *Philadelphia Inquirer*, and the national *CBS Evening News.*

Drexel University is a major university located in Philadelphia PA. Drexel University is among the top 50 private, non-profit, national doctoral/research universities in the US, and has been ranked repeatedly by *U.S. News & World Report* as one of America's Best Colleges. In addition, having one of the oldest and strongest co-op programs, one of the top-ranked online degree programs, and a comprehensive array of colleges and schools, including engineering, media and design arts, biomedical engineering, science and health systems, education, information science, public health, medicine, law, arts & sciences, and business, Drexel has the expertise, depth, and breadth to apply the concept of gaming beyond entertainment to numerous critical industries and fields of study.

"Drexel provided me with an excellent foundation for a career in making games. The multidisciplinary, project-based approach allowed me to experience the game development process from end to end, and gave me the skills to effectively contribute to a team at every point in the pipeline. Thanks to the rigorous curriculum and fantastic professors, I was more than prepared to enter this highly innovative, fast-paced, and demanding industry."

– Will Muto, Program Manager
Xbox LIVE Engagement, Microsoft

"Drexel has continually provided ETC with great co-op talent. We toss the kids into a real multimedia production environment the first day and they are equipped to deliver. Working with RePlay faculty has helped us get talent that is aligned with our needs and kept a steady flow of fresh blood and energy coming into our media lab for years. As lead for our entertainment group, The Ride Works, I am working with Drexel to explore the future of themed entertainment. It's exciting to have access to such a strong and growing program."

– Rob Lloyd, President
& Chief Creative Officer
EnTCo's The Ride Works

Drexel UNIVERSITY RePlay

CONTACT INFO

Drexel University RePlay
3141 Chestnut St.
Philadelphia PA 19104
215.895.1675
www.replay.drexel.edu

GAMESCHOOL DIRECTORY

GAME SCHOOLS ARE GROWING MORE PLENTIFUL BY THE YEAR, AND IT CAN BE DIFFICULT TO KNOW WHAT sort of school is right for you. This directory aims to give you a brief overview of a number of different school types, from art institutions, to traditional universities, to those dedicated to game development.

All the information on these pages was gathered from our sister site www.GameCareerGuide.com, and you should visit pages of any universities in which you may be interested for further information.

So how do you choose? Proximity and cost are the obvious factor, but which school will give you the kind of education you're looking for? One tactic may be to browse student games via Independent Game Festival submissions, or university web sites, and see which of those align with your sensibilities. The types of games that come out of a school aren't a one-to-one indication of the quality of the programs, but it's a start, and can at least give you an idea of the school's games pedigree. Good luck!

NAME	LOCATION	PROGRAMS INCLUDE
3D TRAINING ACADEMY	Bellevue, WA	3D Game Art, Design eXperience Program
3D TRAINING INSTITUTE	New York, NY	3D Foundation Workshop, 3D Project Based Course
ACADEMY COLLEGE	Bloomington, MN	Digital Arts & Design, Computer Animation & Design, Graphic Design, Web Design
ACADEMY OF ART UNIVERSITY	San Francisco, CA	3D Modeling, Background Painting, Character Animation, Games, Storyboard, VFX/Compositing, Visual Development, 2D, 3D, Modeling, Visual Effects, Computer Graphics, Game Environments, Maya, 3D Animation, Character/Tech, Web Design 1
ACADIA UNIVERSITY	Wolfville, Nova Scotia	Game Development
AMERICAN INTERCONTINENTAL UNIVERSITY - DUNWOODY CAMPUS	Atlanta, GA	Game Design & Development, Visual Communication
AMERICAN SENTINEL UNIVERSITY	Aurora, CO	Computer Science, Game Programming
ANGELO STATE UNIVERSITY	San Angelo, TX	Computer Science
ANIMATION ARTS CENTRE - SENECA COLLEGE	Toronto, Ontario	Video Game Art & Design
ANIMATION MENTOR	Emeryville, CA	Advanced Studies in Character Animation
ART CENTER DESIGN COLLEGE - ALBUQUERQUE	Albuquerque, NM	Animation
ART CENTER DESIGN COLLEGE - TUCSON	Tucson, AZ	Animation
ART INSTITUTE OF CALIFORNIA - LOS ANGELES	Santa Monica, CA	Game Art & Design
ART INSTITUTE OF CALIFORNIA - SAN FRANCISCO	San Francisco, CA	Game Art & Design, Visual & Game Programming

abbreviations

R	State residents
AA	Associate of Arts
AAS	Associate of Applied Science
BA	Bachelor of Arts
BCS	Bachelor of Computer Science
BFA	Bachelor of Fine Arts
BGA	Bachelor of Game Art
BS(c)	Bachelor of Science
MA	Master of Arts
MFA	Master of Fine Arts
MS(c)	Master of Science
PhD	Doctor of Philosophy

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
Certificate/Diploma	Varies	–	–	12:1	425.943.7670	www.3dtrainingacademy.com
Certificate/Diploma	$4,650 per full program	–	–	6:1	877.RING.3DTi	www.3dtraining.com
Certificate/Diploma, Associate's	Varies	Yes	Yes	–	952.851.0066	www.academycollege.edu
MA/MFA, Associate's, Online Programs, BFA	$670 per unit	Yes	Yes	15:1	800.544.2787	www.academyart.edu
BCS	$6,652 per Year	Yes	Yes	25:1	902.585-1331	http://cs.acadiau.ca
BFA	Varies	Yes	Yes	–	404.965.8180	http://aiudunwoody.com
BSc	$855 per course	Yes	Yes	15:1	866.922.5690	www.americansentinel.edu
BSc	$203 per unit	–	Yes	25:1	325.942.2101	www.angelo.edu
Certificate/Diploma	Varies	–	–	–	416.491.5050 x3850	http://aac.senecac.on.ca
Certificate/Diploma	$17,875 per full program	–	–	14:1	877.326.4628	www.animationmentor.com
BA/BGA	$680 per unit	Yes	Yes	–	505.254.7575	www.theartcenter.edu
BA/BGA	$710 per unit	Yes	Yes	–	520.325.0123	www.theartcenter.edu
BSc	$21,856 per year	Yes	Yes	19:1	888.646.4610	www.artinstitutes.edu/losangeles
BA/BGA, BFA	$26,432 per year	Yes	Yes	16:1	888.493.3261	www.aicasf.aii.edu

GAMESCHOOL DIRECTORY

NAME	LOCATION	PROGRAMS INCLUDE
ART INSTITUTE OF CALIFORNIA - ORANGE COUNTY	Santa Ana, CA	Game Art & Design, Media Arts & Animation, Visual & Game Programming
ART INSTITUTE OF LAS VEGAS	Henderson, NV	Game Art & Design
ART INSTITUTE OF PITTSBURGH	Pittsburgh, PA	Game Art & Design, Media Arts & Animation, Interactive Media Design, Entertainment Design
ART INSTITUTE OF PORTLAND	Portland, OR	Game Art & Design, Visual & Game Programming
ART INSTITUTE OF SEATTLE	Seattle, WA	Media Arts & Animation, Animation Art & Design, Interactive Media Design, Game Art & Design
AUSTIN COMMUNITY COLLEGE	Austin, TX	Visual Communication, Graphic Design, Graphic Arts Technology, Interactive/Web Design, Game Development, Programming
BAKER COLLEGE ONLINE	Flint, MI	Computer Science, Game Software Development
BECKER COLLEGE	Worcester, MA	Game Design, Game Development & Programming, Game Design
BELLEVUE COMMUNITY COLLEGE	Bellevue, WA	Digital Gaming , Gaming Graphics
BERKELEY DIGITAL FILM SCHOOL	Berkeley, CA	Digital Film
BLOOMFIELD COLLEGE	Bloomfield, NJ	Game Design, Game Programming, Animation, Graphics, Music Technology
BROWN COLLEGE	Mendota Heights, MN	Visual Communications - Multimedia or Graphic Design Emphasis
BRYAN COLLEGE	Springfield, MO	Gaming and Robotics
CALIFORNIA INSTITUTE OF THE ARTS	Valencia, CA	Character Animation, Experimental Design
CALIFORNIA STATE POLYTECHNIC UNIVERSITY - POMONA	Pomona, CA	Computer Science
CAMDEN COUNTY COLLEGE	Blackwood, NJ	Game Design & Development, Computer Graphics, Computer Science
CAÑADA COLLEGE - REDWOOD CITY, CA	Redwood City, CA	Multimedia Art & Technology
CENTER FOR DISTANCE EDUCATION	Sydney, Nova Scotia	3D Game Artist, 3D Animation, 3D Advanced: Character Animation
CENTRE FOR ARTS AND TECHNOLOGY - HALIFAX CAMPUS	Kelowna, British Columbia	3D Animation, 3D Game Animation, Audio Engineering, Digital Filmmaking, Graphic and Digital Media Design, Event and Talent Management

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
BSc	$483 per unit	Yes	Yes	20:1	888.549.3055	www.artinstitutes.edu/orangecounty
BSc	$414 per unit	Yes	Yes	20:1	800.833.2678	www.artinstitutes.edu/lasvegas
BSc	$419 per unit	Yes	Yes	25:1	412.291.6272	www.artinstitutes.edu/pittsburgh
BSc, BFA	$23,880 per year	Yes	Yes	–	800.616.2473	www.artinstitutes.edu/portland
BA/BGA, Associate's	$416 per unit	Yes	Yes	19:1	800.275.2471	www.artinstitutes.edu/seattle
Certificate/Diploma	$1,500 per semester	Yes	Yes	12:1	512.363.3795	www.austincc.edu/viscom
BCS	$185 per unit	Yes	Yes	13:1	888.211.8915	www.baker.edu
BA/BGA	$12,000 per semester	Yes	Yes	12:1	508.373.9731	www.becker.edu/gamedev
Certificate/Diploma	Varies	–	–	–	425.564.1000	www.bcc.ctc.edu
Certificate/Diploma	Varies	–	–	–	510.549.3456	http://berkeleydigital.com/
BG/BGA, Certificate/Diploma	$10,000 per semester	Yes	Yes	15:1	973.748.9000	campus.bloomfield.edu/cat/gamdev.asp
BSc	Varies	–	–	–	888.574.3777	www.browncollege.edu
Associate's, Certificate/ Diploma	Varies	Yes	Yes	–	417.862.5700	www.bryancolleges.edu/ gaming-robotics-specialist.asp
MA/MFA and BFA	$29,300 per year	Yes	Yes	7:1	661.255.1050	www.calarts.edu
MA/MFA and BSc	Varies	–	–	–	909.869.3697	www.csupomona.edu/ffcs
Associate's	$92 per unit	Yes	Yes	–	856.227.7200	www.camdencc.edu
Associate's	$20 per unit	Yes	Yes	15:1	650.306.3201	www.canadacollege.edu/multimedia
Certificate/Diploma	Varies	Yes	Yes	1:1	866.567.3010	www.cd-ed.com
Certificate/Diploma	Varies	Yes	Yes	–	250.860.ARTS	www.digitalartschool.com

NAME	LOCATION	PROGRAMS INCLUDE
CENTRE FOR DIGITAL MEDIA - MASTERS OF DIGITAL MEDIA PROGRAM	Vancouver, British Columbia	Masters of Digital Media Program
CENTRE NAD - NATIONAL ANIMATION AND DESIGN CENTRE	Montreal, Quebec	Design and 3D Animation for Video Games, 3D Animation and Visual Effects for Film and Television, Digital Compositing, Certificate in 3D Animatin and Digital Design
CHAPMAN UNIVERSITY EXTENDED EDUCATION VIDEO GAME CERTIFICATE PROGRAM	Orange, CA	Video Gaming Certificate Program
COGSWELL POLYTECHNICAL COLLEGE	Sunnyvale, CA	Digital Art & Animation, Digital Audio Technology, Digital Arts Engineering, Software Engineering, Computer Engineering
COLLEGE FOR CREATIVE STUDIES	Detroit, MI	Entertainment Arts
COLLEGE OF LAKE COUNTY	Grayslake, IL	Computer Information Technology
COLLINS COLLEGE	Phoenix, AZ	Game Design, Game Art
COLUMBIA COLLEGE CHICAGO	Chicago, IL	Digital Media Technology, Audio for Visual Media, Game Design, Interactive Arts & Media

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
MA/MFA	$10,500 per semester	Yes	Yes	4:1	778.370.1031	www.mdm.gnwc.ca
Certificate/Diploma	Varies	Yes	Yes	22:1	1-514-288-3447	www.nadcentre.com
Certificate/Diploma	$400 per unit	Yes	–	15:1	714.744.2125	www.chapman.edu/exed/gaming
BA/BGA, BSc	$8,496 per semester	Yes	Yes	12:1	800.264.7955	www.cogswell.edu
BFA	Varies	Yes	Yes	11:1	800.952.ARTS	www.collegeforcreativestudies.edu
Certificate/Diploma, Associate's	$90 per semester	Yes	Yes	19:1	847.543.2041	www.clcillinois.edu/programs/cit/index.asp?gaming
BA/BGA	$6,200 per semester	Yes	Yes	30:1	800.876.7070	www.collinscollege.edu
BA/BGA	$17,104 per year	Yes	Yes	20:1	312.663.1600	www.iam.colum.edu www.game.colum.edu

NAME	LOCATION	PROGRAMS INCLUDE
CORNELL UNIVERSITY	Ithica, NY	Game Design Minor
DELAWARE COUNTY COMMUNITY COLLEGE	Media, PA	Video Game Design and Development
DEPAUL UNIVERSITY	Chicago, IL	Game Development (Programming, Production & Design), Animation, Computer Graphics Programming, Interactive Media, Digital Cinema, Computer Science
DESALES UNIVERSITY	Center Valley, PA	Computer Science - Game Programming Track
DEVRY UNIVERSITY - DALLAS	Irving, TX	Game and Simulation Programming
DEVRY UNIVERSITY - LONG BEACH	Long Beach, CA	Game and Simulation Programming
DEVRY UNIVERSITY - OAKBROOK TERRACE, IL	Oakbrook Terrace, IL	Game and Simulation Programming
DIGIPEN INSTITUTE OF TECHNOLOGY	Redmond, WA	Real-Time Interactive Simulation, Compute Engneering, Computer Science, Production Animation, Game Design
DIGITAL MEDIA ARTS COLLEGE	Boca Raton, FL	Computer Animation, Special Effects Animation

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
BSc, BA/BGA	$18,150 per semester	Yes	Yes	9:1	607.255.9196	http://gdiac.cis.cornell.edu
Associate's	Varies	–	–	–	610.359.5365	www.dccc.edu
MSc, MA/MFA, BSc, BA/BGA	$27,055 per year	Yes	Yes	16:1	312.362.8381	http://gamedev.depaul.edu
BSc	$26,000 per year	Yes	Yes	17:1	610.282.1100	www.desales.edu
BSc	Varies	–	–	–	866.338.7934	www.devry.edu/locations/campuses/loc_irvingcampus.jsp
BSc	Varies	Yes	Yes	20:1	800.597.1333	www.devry.edu
BSc	Varies	–	–	–	866.338.7934	www.devry.ed
BSc, MSc, BFA, BA/BGA	$476 per unit	Yes	Yes	13:1	425.558.0299	www.digipen.edu
MA/MFA and BFA	Varies	–	–	–	866.255.DMAC	www.dmac.edu

GAMESCHOOL DIRECTORY

NAME	LOCATION	PROGRAMS INCLUDE
DREXEL UNIVERSITY, ANTOINETTE WESTPHAL COLLEGE OF MEDIA ARTS AND DESIGN	Philadelphia, PA	Digital Media, Film & Video, Graphic Design, Photography, Screenwriting & Playwriting
EMILY CARR UNIVERSITY OF ART + DESIGN	Vancouver, British Columbia	Illustration, Animation
EXPRESSION COLLEGE FOR DIGITAL ARTS	Emeryville, CA	Animation and Visual Effects, Game Art and Design, Motion Graphic Design, Sound Arts
FLASHPOINT - THE ACADEMY OF MEDIA ARTS & SCIENCES	Chicago, IL	Game Development, Visual Effects/Animation, Recording Arts, Film/Broadcast
FULL SAIL UNIVERSITY	Winter Park, FL	Game Development, Computer Animation, Game Art, Game Design
FUTUREPOLY	Bellevue, WA	3D Modeling for Games, Texturing for Games, Digital Painting, Concept Art, Character Modeling with Zbrush
GAME DESIGN SCHOOL IN VANCOUVER, BC CANADA	Vancouver, British Columbia	Audio Engineering & Production, Film & Digital Arts, Film & Music Business, Game Design & 3D Animation
GEMINI SCHOOL OF VISUAL ARTS & COMMUNICATION	Cedar Park, TX	Commercial Art & Illustration
GNOMON ONLINE	Hollywood, CA	Online Classes
GREAT EASTERN TECHNOLOGY	Woburn, MA	Visual Computing Training
GUILDHALL AT SMU	Plano, TX	Art Creation, Level Design, Software Development
HAGERSTOWN COMMUNITY COLLEGE	Hagerstown, MD	Simulation and Digital Entertainment
HARPER COLLEGE CE	Schaumburg, IL	Flash Game Designer CE Certificate
HIGH POINT UNIVERSITY	High Point, NC	Game and Interactive Media Design
HOUSTON COMMUNITY COLLEGE	Houston, TX	Digital Gaming (For Artists and Programmers)
ILLINOIS INSTITUTE OF ART - CHICAGO	Chicago, IL	Game Art & Design, Media Arts & Animation, Digital Media Production, Visual Efffects & Motion Graphics
ILLINOIS INSTITUTE OF ART - SCHAUMBURG	Schaumburg, IL	Game Art & Design, Interactive Media Design, Media Arts & Animation, Visual Effects and Motion Graphics
INTERNATIONAL ACADEMY OF DESIGN & TECHNOLOGY - MICHIGAN	Troy, MI	Game Production
INTERNATIONAL ACADEMY OF DESIGN & TECHNOLOGY - ONTARIO	Ontario	Video Game Design and Development

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
MSc, Certificate/Diploma, BSc	$28,300 per year	Yes	Yes	15:1	215.895.1675	http://drexel.edu/westphal/academics/undergraduate/digitalmedia
BA/BGA, MA/MFA	$3,500 per year	Yes	Yes	18:1	604-844-3897	www.ecuad.ca
BSc	Varies	Yes	Yes	–	877.833.8800	www.expression.edu
Certificate/Diploma	$25,000 per year	–	Yes	20:1	312.332.0707	www.flashpointacademy.com
BSc, MSc	Varies	Yes	Yes	9:1	407.679.0100	www.fullsail.edu
Certificate/Diploma	$3,200 per semester	–	–	11:1	206.331.1573	www.futurepoly.com
Certificate/Diploma	Varies	Yes	Yes	–	604.873.4853	http://pacificav.com/gamedesign/gd.html
Certificate/Diploma	$17,000 per year	–	Yes	24:1	512.249.1237	www.geminischool.com
Certificate/Diploma	$1,650 per course	Yes	Yes	12:1	323.466.6663	www.gnomononline.com
Certificate/Diploma	Varies	–	–	–	800.875.0025	www.get.com
MIT, Certificate/Diploma	$51,000 per full program	Yes	Yes	12:1	214.768.9950	http://guildhall.smu.edu
Certificate/Diploma, Associate's and AAS	$1,293 per semester	–	Yes	18:1	301.790.2800 x361	www.hagerstowncc.edu
Certificate/Diploma	$1,400 per full program	–	–	10:1	847.925.6066	http://goforward.harpercollege.edu/page.cfm?p=2469
BA/BGA	$33,000 per year	Yes	Yes	14:1	336.841.9174	www.highpoint.edu
AAS, Certificate/Diploma	Varies	–	–	–	713.718.5728	swc2.hccs.edu/digigame
BFA	$22,800 per year	Yes	Yes	15:1	312.280.3500	www.ilic.aii.edu
BFA	$24,000 per year	Yes	Yes	19:1	800.314.3450	www.artinstitutes.edu/schaumburg
BA/BGA and BFA	Varies	Yes	Yes	–	248.457.2700	www.iadtdetroit.com
–	Varies	–	–	–	800.361.6664	www.iaod.com

NAME	LOCATION	PROGRAMS INCLUDE
INTERNATIONAL ACADEMY OF DESIGN AND TECHNOLOGY - CHICAGO	Chicago, IL	Visual Communications (Game Design, Video & Animation Production, Multimedia & Web Design)
ITP AT NEW YORK UNIVERSITY	New York, NY	Interaction Design
ITT TECH - GREEN BAY, WISCONSIN	Green Bay, WI	Digital Entertainment & Game Design
ITT TECH - GREENVILLE, SOUTH CAROLINA	Greenville, SC	Digital Entertainment & Game Design, Visual Communications
ITT TECH - LOUISVILLE, KENTUCKY	Louisville, KY	Digital Entertainment & Game Design
ITT TECH - ARNOLD, MISSOURI	Arnold, MO	Digital Entertainment & Game Design
JOHNSON COUNTY COMMUNITY COLLEGE	Overland Park, KS	Computer Game Development
KAPLAN UNIVERSITY	Fort Lauderdale, FL	Web Design & Animation for Gaming
KEISER UNIVERSITY	Fort Lauderdale, FL	Video Game Design

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
BFA	$22,400 per year	–	–	11:1	888.318.6111	www.iadtchicago.edu
MPS	$45,000 per year	Yes	Yes	5:1	212.998.1882	http://itp.nyu.edu
BSc, Associate's	Varies	–	–	–	920.662.9000	http://itt-tech.edu
BSc, Associate's	Varies	Yes	Yes	–	864.288.0777	http://itt-tech.edu
BSc	Varies	–	–	–	502.327.7424	www2.itt-tech.edu/campus/courses.cfm?prog_id=2171
BSc	Varies	Yes	Yes	–	636.464.6600 x150	www.itt-tech.edu/campus/school.cfm?lloc_num=91
Associate's	Varies	Yes	Yes	–	916.465.4900	www.jccc.edu
BSc, BA/BGA, and Associate's	$4,900 per semester	Yes	Yes	–	800.817.8272	www.online.kaplanuniversity.edu
Associate's	Varies	Yes	Yes	5:1	954.776.4456	www.keiseruniversity.edu

NAME	LOCATION	PROGRAMS INCLUDE
LIVING ARTS COLLEGE	Raleigh, NC	Animation & Game Design
LOS ANGELES FILM SCHOOL - GAME PRODUCTION	Hollywood, CA	Game Production
MADISON MEDIA INSTITUTE - COLLEGE OF MEDIA ARTS	Madison, WI	Game Art & Animation, Recording & Music Technology, Game Art & Animation, Video & Motion Graphics
MASSACHUSETTS INSTITUTE OF TECHNOLOGY (MIT)	Cambridge, MA	Comparative Media Studies, Electrical Engineering and Computer Science, Sloan School of Management
MAX THE MUTT ANIMATION SCHOOL	Toronto, Ontario	Concept Art, Classical & Computer Animation & Production
MESA COMMUNITY COLLEGE	Mesa, AZ	Applied Science in Game Technology
MICHIGAN STATE UNIVERSITY	East Lancing, MI	Game Design & Development Specialization, Serious Game Design
MILWAUKEE AREA TECHNICAL COLLEGE	Milwaukee, WI	Animation
MISSOURI STATE UNIVERSITY - WEST PLAINS	West Plains, MO	Computer Graphics & Programming
MONTGOMERY COLLEGE	Rockville, MD	Computer Gaming and Simulation (Art & Animation, Programming, and Production & Design tracks), Internet Gaming and Simulation
MT. SIERRA COLLEGE	Monrovia, CA	Game Arts and Design
NATIONAL UNIVERSITY - SCHOOL OF MEDIA & COMMUNICATION	La Jolla, CA	Video Game Production and Design, Digital Entertainment and Interactive Arts
NBCC MIRAMICHI	Miramichi, New Brunswick	Media Studies (Art Fundamentals), Animation and Graphics, Electronic Game - 3D Graphics, Electronic Game - Design (Programming)
NEW ENGLAND INSTITUTE OF ART	Brookline, MA	Media Arts & Animation
NEW ENGLAND INSTITUTE OF TECHNOLOGY	Warwick, RI	Game Development and Simulation Programming Technology, Software Engineering Technology, Video and Audio Production, Digital Recording Arts, Graphics, Multimedia and Web Design
NEW JERSEY INSTITUTE OF TECHNOLOGY	Newark, NJ	Information Technology
NHTI CONCORD COMMUNITY COLLEGE	Concord, NH	Animation & Graphic Game Programming
NORTH CAROLINA STATE UNIVERSITY - COLLEGE OF DESIGN	Raleigh, NC	3D Animation, Game Design, Interactive Design, Multimedia & Digital Imaging, Ideation & Illustration
NORTHEASTERN UNIVERSITY	Boston, MA	Interactive Design, Game Design, Digital Video, 3D Animation

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
BA/BGA	$77,040 per full program	Yes	Yes	20:1	800.288.7442	www.living-arts-college.edu
Associate's	$37,900 per full program	Yes	Yes	–	877.952.3456	www.lafilm.edu
Associate's	$470 per unit	Yes	Yes	16:1	800.236.4997	www.madisonmedia.edu
PhD, MSc, BSc	$34,986 per year	Yes	Yes	7:1	617.253.3400	http://web.mit.edu/admissions
Certificate/Diploma	Varies	–	–	–	416.703.6877	www.maxthemutt.com
Associate's, Certificate/ Diploma	$225 per course	Yes	Yes	15:1	480.461.7463	www.mc.maricopa.edu/fflybbert/ studio180/degrees.html
BA/BGA, MA/MFA	$10,264 per semester	–	Yes	15:1	517.353.5497	http://seriousgames.msu.edu
Associate's	$325 per course	Yes	Yes	20:1	414.297.MATC	www.matc.edu
Associate's	$102 per unit	Yes	Yes	18:1	417.255.7298	www.wp.missouristate.edu/cgp
Certificate/Diploma, Associate's, MCS, AA	Varies	Yes	Yes	–	240.567.5000	www.studygaming.com
BSc	Varies	–	–	–	626.873.2144	www.mtsierra.edu
MFA, BA/BGA	Varies	–	–	–	858.642.8434	www.nu.edu/academics/schools/ somc.html
Certificate/Diploma	$2,600 per year	–	Yes	–	506.778.6000	www.nbcc.ca/miramichi
BCs	Varies	–	–	–	800.903.4425	www.artinstitutes.edu/boston
Associate's, BSc	$16,200 per year	Yes	Yes	15:1	401.467.7744	www.neit.edu
BSc	$4,850 per semester	Yes	Yes	20:1	973.596.5764	www.njit.edu
Associate's	$175 per unit	Yes	Yes	20:1	603.271.7757	www.nhti.edu/academics/ academicprograms/degaggp.html
MA/MFA, BFA	$14,454 per year	Yes	Yes	18:1	919.515.3876	www.ncsudesign.org
Certificate/Diploma	$514 per unit	Yes	Yes	12:1	617.373.2400	www.northeastern.edu/cps/digitalmedia

NAME	LOCATION	PROGRAMS INCLUDE
NORTHERN OKLAHOMA COLLEGE	Tonkawa, OK	3D Animation & Post-Production
OHIO UNIVERSITY	Athens, OH	Digital Media: Special Effects, Games, and Animation
OKLAHOMA CITY COMMUNITY COLLEGE	Oklahoma City, OK	Computer Aided Technology: Game Design Emphasis
OKLAHOMA PANHANDLE STATE UNIVERSITY	Goodwell, OK	Game Art Design, Computer Graphics
OREGON3D, INC.	Portland, OR	–
OTIS COLLEGE OF ART AND DESIGN - DIGITAL MEDIA DEPARTMENT	Los Angeles, CA	Game Design, Animation, Interactive Design, Motion Graphics, Visual Effects
PARSONS THE NEW SCHOOL OF DESIGN - SCHOOL OF ART, MEDIA AND TECHNOLOGY	New York, NY	Design & Technology
PIEDMONT COMMUNITY COLLEGE	Roxboro, NC	–
PINNACLE COLLEGE	Alhambra, CA	Audio for Games and Interactive Media

DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
AAS	$3,720 per year	Yes	Yes	5:1	580.628.6458	www.north-ok.edu/ics
MA/MFA, BSc	$2,909 per semester	Yes	Yes	16:1	740.593.4870	www.tcomschool.ohiou.edu
Certificate/Diploma	$54 per unit	Yes	Yes	20:1	405.682.1611 x7498	http://catblog.occc.edu
B.Tech, BA/BGA	$249 per course	Yes	Yes	18:1	580.349.1469	www.opsu.edu
–	Varies	–	–	–	866.626.9100	www.oregon3d.com
BFA	$30,414 per year	Yes	Yes	10:1	310.665.6800	www.otis.edu
MA/MFA, BFA	$31,000 per year	Yes	Yes	15:1	800.252.0852	http://cdt.parsons.edu
–	Varies	–	–	–	336.694.5707	www.piedmontcc.edu
Associate's	$38,000 per full program	Yes	Yes	–	877.206.6206	www.pinnaclecollege.edu

NAME	LOCATION	PROGRAMS INCLUDE
RASMUSSEN COLLEGE	Multiple Locations	Game and Simulation Production, Information Systems Management, Multimedia Technologies, Microsoft and Cisco
RENSSELAER POLYTECHNIC INSTITUTE	Troy, NY	Game and Simulation Arts & Sciences
RINGLING COLLEGE OF ART AND DESIGN	Sarasota, FL	Computer Animation, Graphic & Interactive Communication, Game Art & Design
ROCHESTER INSTITUTE OF TECHNOLOGY	Rochester, NY	Game Design & Development, Information Technology, Computer Science, Computer Graphics Design, New Media Interactive Development, New Media Design and Imaging, 3D Digital Graphics
SAN FRANCISCO SCHOOL OF DIGITAL FILMMAKING	San Francisco, CA	–
SANTA ANA COLLEGE	Santa Ana, CA	3D Animation
SANTA MONICA COLLEGE - ACADEMY OF ENTERTAINMENT TECHNOLOGY	Santa Monica, CA	Animation, Game Development, Post Production, Visual Effects
SAVANNAH COLLEGE OF ART AND DESIGN	Savannah, GA	Animation, Interactive Game Design & Motion Graphics, Sound Design, Visual Effects
SCHOOL OF VISUAL ARTS	New York, NY	Computer Art, Computer Animation, and Visual effects
SEATTLE CENTRAL COMMUNITY COLLEGE	Seattle, WA	3D Animation/Design & Gaming Program
SENECA COLLEGE - GAME ART & ANIMATION	Toronto, Ontario	3D Animation and 3D Gaming
SERIOUS GAME DESIGN INSTITUTE	Santa Barbara, CA	Serious Game and Simulation Design
SHAWNEE STATE UNIVERSITY	Portsmouth, OH	Digital Simulation and Gaming Engineering Technology, Gaming and Simulation Development Arts
SIMON FRASER UNIVERSITY - SCHOOL OF INTERACTIVE ARTS AND TECHNOLOGY (SIAT)	Surrey, British Colombia	Media Arts, Design & Informatics
SOUTHERN POLYTECHNIC STATE UNIVERSITY	Marietta, GA	Computer Game Design and Development, Computer Science, Information Technology, Software Engineering
SPRINGFIELD COLLEGE	Springfield, MA	Concentration in Game Programming within Computer & Information Sciences Major
ST. PETERSBURG COLLEGE-SEMINOLE CAMPUS	Seminole, FL	Digital Media Production, Video Game Foundations, Digital Media Video Production
STONY BROOK UNIVERSITY	Stony Brook, NY	Computer Science Specialization in Game Programming

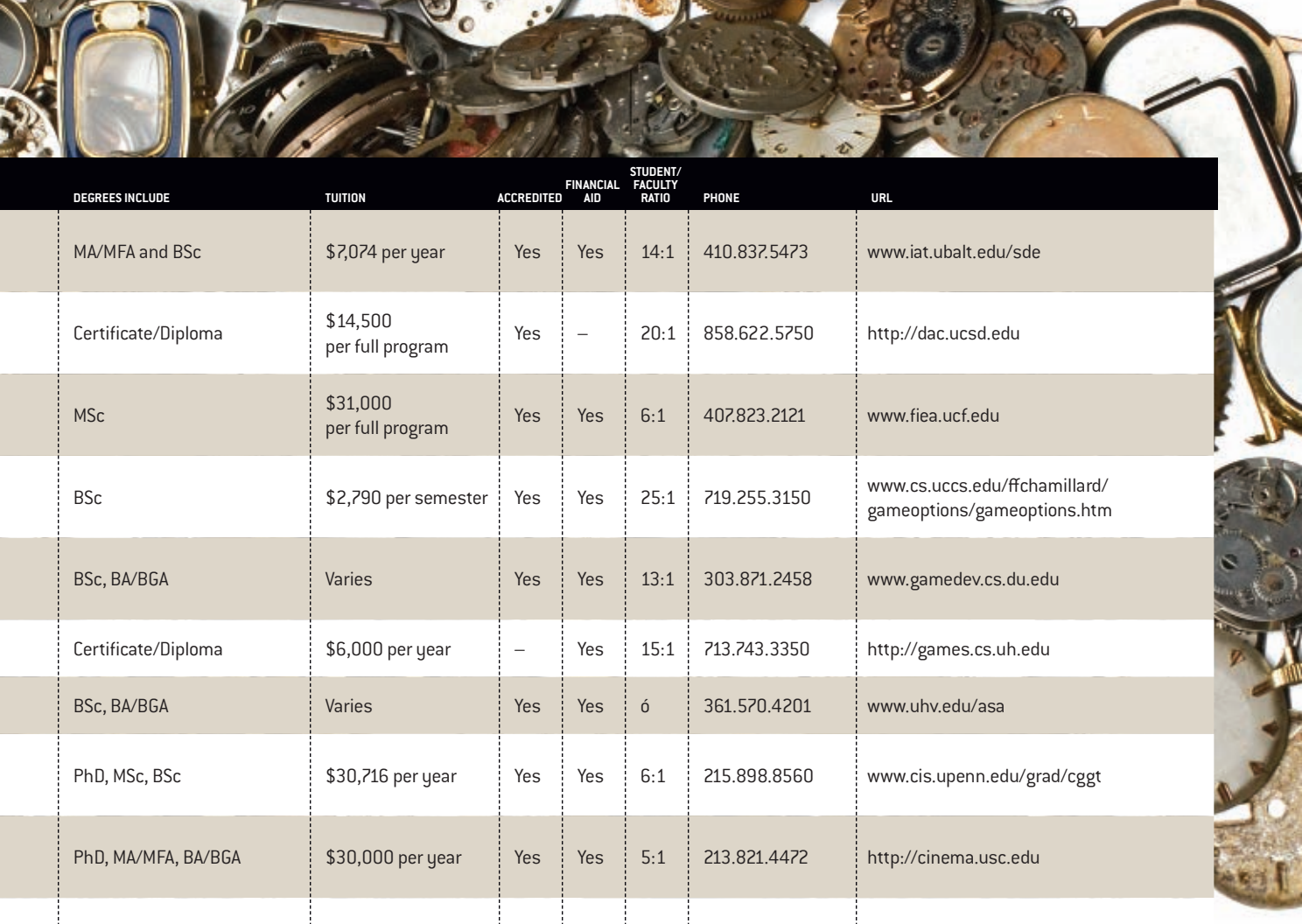
DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
BSc, Associate's, Certificate/ Diploma	Varies	Yes	Yes	–	888.549.6755	www.rasmussen.edu/design/programs/game-simulation-production.asp
BSc	Varies	Yes	Yes	–	518.276.6575	www.gsas.rpi.edu
BFA	$14,900 per semester	Yes	Yes	13:1	800.255.7695	www.ringling.edu
MSc, BSc, MA/MFA	$28,866 per year	Yes	Yes	13:1	585.475.6631	www.rit.edu
–	Varies	–	–	–	877.292.4200	http://sfdigifilm.com
Certificate/Diploma	$20 per unit	Yes	Yes	18:1	714.564.6741	http://ext.sac.edu/academic_progs/art/3d
Certificate/Diploma, Associate's	$20 per unit	Yes	Yes	18:1	310.434.3700	http://academy.smc.edu
Certificate/Diploma, BFA, MA/MFA	$22,950 per year	Yes	Yes	16:1	800.869.7223	www.scad.edu
BFA, MFA	$13,400 per semester	Yes	Yes	9:1	212.592.2116	www.sva.edu
Certificate/Diploma	$395 per course	Yes	–	10:1	206.587.5448	www.learnatcentral.org
Certificate/Diploma, MSc	Varies	Yes	Yes	15:2	416.491.5050 x3850	www.senecagaming.ca
–	$60 per course	Yes	Yes	28:1	805.965.0581	http://sgdi.sbcc.edu
BSc, BA/BGA	$2,195 per semester	Yes	Yes	20:1	740.351.3113	http://ssugames.org and http://www.shawnee.edu
BSc, BA/BGA	$465 per course	Yes	Yes	–	778.782.7474	www.siat.sfu.ca
BSc, MSc, Certificate/Diploma	$2,100 per semester	Yes	Yes	20:1	678.915.4982	http://games.spsu.edu
BSc	$25,100 per year	Yes	Yes	13:1	413.748.3313	www.springfieldcollege.edu
Certificate/Diploma, Associate's	Varies	Yes	Yes	–	727.394.6111	www.spcollege.edu/se/digital_media/index.htm
BSc	$4,970 per year	Yes	Yes	19:1	631.632.8470	www.cs.stonybrook.edu

NAME	LOCATION	PROGRAMS INCLUDE
TEXAS STATE TECHNICAL COLLEGE	Waco, TX	Game & Simulation, Game Programming & Design
THE ACADEMY OF ENTERTAINMENT AND TECHNOLOGY AT SANTA MONICA COLLEGE	Santa Monica, CA	Game Design, Animation
THE ART INSTITUTE OF VANCOUVER	Vancouver, British Columbia	Game Art & Design, Visual & Game Programming
THE COLLEGE OF WESTCHESTER	White Plains, NY	Multimedia Development & Management
THE PIXELYARD SCHOOL OF ART	San Diego, CA	Game Art
THE UNIVERSITY OF THE ARTS	Philadelphia, PA	Multimedia
TRIOS COLLEGE	Mississauga, Ontario	Video Game Design and Development
TYLER JUNIOR COLLEGE	Tyler, TX	Gaming and Simulation Development (Programming and Graphics)
UNIVERSITY OF ADVANCING TECHNOLOGY	Tempe, AZ	Game Design, Game Programming, Software Engineering, Game Art & Animation, Game Production

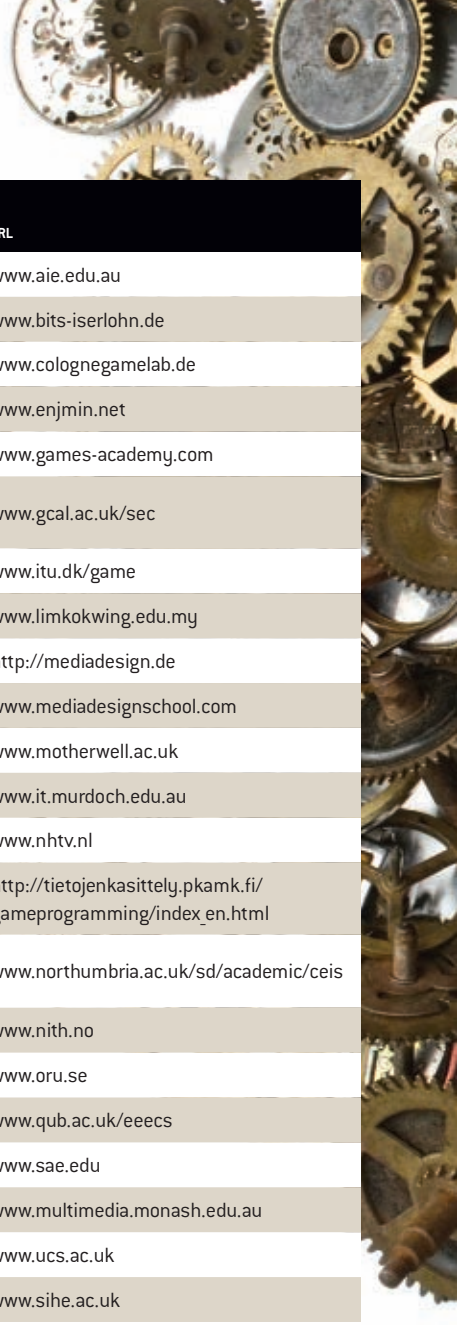
DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
Associate's, BSc	$1,400 per semester	Yes	Yes	15:1	800.792.8784 x4862	www.waco.tstc.edu
Certificate/Diploma, Associate's	$20 per unit	Yes	Yes	24:1	310.434.3700	http://academy.smc.edu
Certificate/Diploma	Varies	Yes	Yes	20:1	800.661.1885	www.artinstitutes.edu/vancouver
Associate's	Varies	Yes	Yes	16:1	800.333.4924	www.cw.edu
Certificate/Diploma	Varies	–	Yes	4:1	858.427.1616	www.thepixelyard.com
BFA	$27,220 per year	Yes	Yes	9:1	800.616.ARTS	www.uarts.edu
Certificate/Diploma	Varies	Yes	Yes	20:1	905.814.7212	www.getintothegame.ca
Associate's	$1,200 per semester	Yes	Yes	15:1	903.510.2348	www.tjc.edu
BA/BGA, BSc, MA/MFA, MSc, Associate's	$8,400 per semester	Yes	Yes	14:1	877.UAT.GEEK	www.uat.edu

GAMESCHOOL DIRECTORY

NAME	LOCATION	PROGRAMS INCLUDE
UNIVERSITY OF BALTIMORE	Baltimore, MD	Simulation and Digital Entertainment, Interaction Design & Information Architechture
UNIVERSITY OF CALIFORNIA SAN DIEGO - EXTENSION DIGITAL ARTS CENTER	San Diego, CA	Casual Game Development, Digital Graphics & Web Design, Video, Sound & Motion Graphics
UNIVERSITY OF CENTRAL FLORIDA (FIEA)	Orlando, FL	Programming, Art, and Production Tracks
UNIVERSITY OF COLORADO, COLORADO SPRINGS	Colorado Springs, CO	Game Design and Development
UNIVERSITY OF DENVER	Denver, CO	Animation and Game Development, Computer Science, Digital Media Studies, Electronic Media Arts Design
UNIVERSITY OF HOUSTON	Houston, TX	Interactive Game Development
UNIVERSITY OF HOUSTON - VICTORIA	Victoria, TX	Digital Gaming and Simulation
UNIVERSITY OF PENNSYLVANIA	Philadelphia, PA	Computer Graphics & Game Technology, Digital Media & Design, Human Modeling & Simulation
UNIVERSITY OF SOUTHERN CALIFORNIA - INTERACTIVE MEDIA DIVISION	Los Angeles, CA	Interactive Entertainment, Interactive Media
UNIVERSITY OF TEXAS AT DALLAS - ARTS AND TECHNOLOGY PROGRAM	Richardson, TX	Arts and Technology
UNIVERSITY OF UTAH - ENTERTAINMENT ARTS AND ENGINEERING MASTERS STUDIO	Salt Lake City, UT	Entertainment Art and Engineering Masters Studio
UNIVERSITY OF WASHINGTON - BOTHELL	Bothell, WA	Computing and Software Systems
UNIVERSITY OF WASHINGTON - PROFESSIONAL AND CONTINUING EDUCATION	Seattle, WA	Game Development, 3D Animation for Games & Digital media, Virtual Worlds
UNIVERSITY OF WATERLOO	Waterloo, Ontario	Computer Science & Computer Engineering
VANCOUVER COLLEGE OF ART & DESIGN	Vancouver, British Columbia	3D Modeling Animation Art & Design
VANCOUVER FILM SCHOOL	Vancouver, British Columbia	Game Design, 3D Animation, Classical Animation, Digital Character Animation (Maya), Sound Design
VIRGINIA COMMONWEALTH UNIVERSITY	Richmond, VA	Communication Arts, Design
WAKE TECHNICAL COMMUNITY COLLEGE	Raleigh, NC	Simulation & Game Development
WESTCHESTER COM. COLL. PEEKSKILL EXTENSION CENTER - NEW YORK	Peekskill, NY	Design & New Media
WESTWOOD COLLEGE ONLINE	Denver, CO	Game Software Development, Game Art & Design



DEGREES INCLUDE	TUITION	ACCREDITED	FINANCIAL AID	STUDENT/ FACULTY RATIO	PHONE	URL
MA/MFA and BSc	$7,074 per year	Yes	Yes	14:1	410.837.5473	www.iat.ubalt.edu/sde
Certificate/Diploma	$14,500 per full program	Yes	–	20:1	858.622.5750	http://dac.ucsd.edu
MSc	$31,000 per full program	Yes	Yes	6:1	407.823.2121	www.fiea.ucf.edu
BSc	$2,790 per semester	Yes	Yes	25:1	719.255.3150	www.cs.uccs.edu/ffchamillard/gameoptions/gameoptions.htm
BSc, BA/BGA	Varies	Yes	Yes	13:1	303.871.2458	www.gamedev.cs.du.edu
Certificate/Diploma	$6,000 per year	–	Yes	15:1	713.743.3350	http://games.cs.uh.edu
BSc, BA/BGA	Varies	Yes	Yes	ó	361.570.4201	www.uhv.edu/asa
PhD, MSc, BSc	$30,716 per year	Yes	Yes	6:1	215.898.8560	www.cis.upenn.edu/grad/cggt
PhD, MA/MFA, BA/BGA	$30,000 per year	Yes	Yes	5:1	213.821.4472	http://cinema.usc.edu
MA/MFA, BA/BGA	$4,110 per semester	Yes	Yes	33:1	972.883.4331	atec.utdallas.edu
MA/MFA, MSc	$7,500 per semester	Yes	Yes	5:1	801.581.8224	www.eaems.utah.edu
BSc, MSc	Varies	–	–	–	425.352.5000	www.bothell.washington.edu
Certificate/Diploma	$705 per course	Yes	–	25:1	888.469.6499	www.extension.washington.edu
PhD and BSc	$14,982 per year	Yes	Yes	19:1	519.888.4567	www.cs.uwaterloo.ca
Certificate/Diploma	Varies	Yes	Yes	–	See website	www.vancouver.vcad.ca
British Columbia	Varies	Yes	Yes	16:1	604.685.5808	www.vfs.com
Certificate/Diploma	Varies	–	–	–	804.828.0100	www.vcu.edu
Certificate/Diploma, Associate's	Varies	Yes	Yes	–	919.866.5949	http://cet.waketech.edu/sgd/sgd.htm
Certificate/Diploma	Varies	–	–	–	914.606.7359	www.sunywcc.edu/peekskill
BSc	Varies	–	–	–	866.246.8831	www.westwoodonline.edu



INTERNATIONAL SCHOOLS

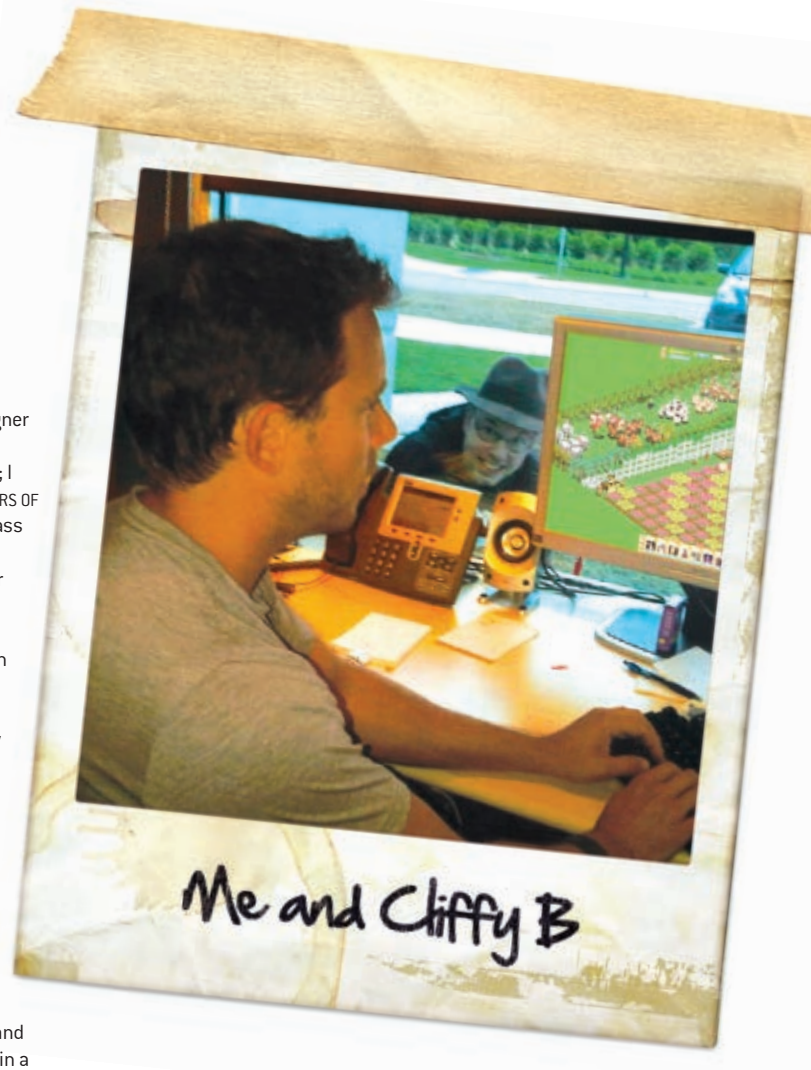
NAME	LOCATION	URL
ACADEMY OF INTERACTIVE ENTERTAINMENT, MELBOURNE	Melbourne, Australia	www.aie.edu.au
BUSINESS AND INFORMATION TECHNOLOGY SCHOOL (BITS)	Iserlohn, Germany	www.bits-iserlohn.de
COLOGNE GAME LAB	Cologne, Germany	www.colognegamelab.de
ENJMIN (NATIONAL SCHOOL OF VIDEO GAME AND INTERACTIVE MEDIA)	AngoulÍme, France	www.enjmin.net
GAMES ACADEMY	Berlin, Germany	www.games-academy.com
GLASGOW CALEDONIAN UNIVERSITY, SCHOOL OF ENGINEERING AND COMPUTING	Glasgow, Scotland	www.gcal.ac.uk/sec
IT UNIVERSITY OF COPENHAGEN	Copenhagen, Denmark	www.itu.dk/game
LIMKOKWING UNIVERSITY COLLEGE OF CREATIVE TECHNOLOGY	Cyberjaya, Malaysia	www.limkokwing.edu.my
MD.H MEDIADESIGN - HOCHSCHULE	Berlin, Germany	http://mediadesign.de
MEDIA DESIGN SCHOOL	Auckland, New Zealand	www.mediadesignschool.com
MOTHERWELL COLLEGE	Motherwell, United Kingdom	www.motherwell.ac.uk
MURDOCH UNIVERSITY	Perth, Australia	www.it.murdoch.edu.au
NHTV BREDA UNIVERSITY OF APPLIED SCIENCE	Breda, Netherlands	www.nhtv.nl
NORTH KARELIA UNIVERSITY OF APPLIED SCIENCES	Joensuu, Finland	http://tietojenkasittely.pkamk.fi/gameprogramming/index_en.html
NORTHUMBRIA UNIVERSITY	Newcastle upon Tyne, United Kingdom	www.northumbria.ac.uk/sd/academic/ceis
NORWEGIAN SCHOOL OF INFORMATION TECHNOLOGY	Oslo, Norway	www.nith.no
ÖREBRO UNIVERSITY	Örebro, Sweden	www.oru.se
QUEEN'S UNIVERSITY BELFAST - SCHOOL OF EEECS	Belfast, United Kingdom	www.qub.ac.uk/eeecs
SAE INSTITUTE SOUTH AFRICA	Cape Town, South Africa	www.sae.edu
SCHOOL OF MULTIMEDIA SYSTEMS	Berwick, Australia	www.multimedia.monash.edu.au
SUFFOLK SCHOOL OF ARTS, UNIVERSITY CAMPUS SUFFOLK	Ipswich, United Kingdom	www.ucs.ac.uk
SWANSEA METROPOLITAN UNIVERSITY - SCHOOL OF DIGITAL MEDIA	Swansea, United Kingdom	www.sihe.ac.uk
TAFE NSW - WOLLONGONG WEST	Wollongong, Australia	www.illawarra.tafensw.edu.au
THE GAME ASSEMBLY	Malmö, Sweden	www.thegameassembly.com
UNIVERSIDADE ANHEMBI MORUMBI	So Paulo, Brazil	www.anhembi.br
UNIVERSITY OF ABERTAY DUNDEE	Dundee, Scotland	www.abertay.ac.uk
UNIVERSITY OF APPLIED ARTS AND DESIGN	Zurich, Switzerland	http://interaction.hgkz.ch
UNIVERSITY OF APPLIED SCIENCES SALZBURG / FH SALZBURG	Puch bei Hallein, Austria	www.fh-salzburg.ac.at
UNIVERSITY OF GÄVLE	Gävle, Sweden	www.hig.se
UNIVERSITY OF GLAMORGAN	Pontypridd, United Kingdom	www.glam.ac.uk/coursedetails/685/53
UNIVERSITY OF GOTLAND	Visby, Sweden	http://mainweb.hgo.se
UNIVERSITY OF HULL - DEPARTMENT OF COMPUTER SCIENCE	Hull, United Kingdom	www.net.dcs.hull.ac.uk
UNIVERSITY OF OTAGO	Dunedin, New Zealand	www.otago.ac.nz
UNIVERSITY OF UTRECHT	Utrecht, Netherlands	www.uu.nl
ZURICH UNIVERSITY OF THE ARTS	Zurich, Switzerland	www.zhdk.ch

ADVERTISER INDEX

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. 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.

CLIFFY B MINE

A FAN'S PLEA TO CLIFF BLESZINSKI (DON'T TRY THIS AT HOME!)



Dear CliffyB a.k.a. Cliff Bleszinski (ha ha),

Yo dude, it's me, your numero uno fan and future awesome game designer at Epic Games (your studio!!). I've been trying to get in touch with you again, but you never seem to be around when I call. That's cool, though; I know you're busy and probably working a lot of long hours making GEARS OF WAR 3, BULLETSTORM, and who knows what other games as totally kick-ass as possible! And you probably come home late at night really tired and just go right to bed, which is why you don't pick up when I call you after midnight when I think you might be there.

But that's why I'm writing a letter now. I want you to know I'm ready for you to hire me. I've "done my time," formulating game design theories and the like on well-known Internet forums. As stunning and insightful as I know these posts were, I didn't seem to attract your attention right off the bat. So I figured all I needed to do was somehow get you to just take a look at them, and you'd recognize my talent straight away. I just knew that if you read my design ideas, you'd totally understand, man. I mean, I play your games all the time and I just feel like, yes! This guy gets me, and I get him! Him and me, we were destined to make games together!

And I know you'll get this message for sure, because nobody can resist clam chowder! Am I right or am I right? That chowder I sent you is homemade, by the way—it's an old family recipe that's been passed down and it just can't be beat, at least according to allrecipes.com. Hopefully none of it has spilled in transit and you've carefully fished this letter out of the vat I had shipped over via FedEx, and are reading it now in between big delicious heaping spoonfuls. (I put it in a plastic bag. Clever, right?) Don't hog it all to yourself—share some with the crew too! You cheeky bastard! Ha ha!

Anyway, I do hope my gift goes a way toward apologizing for the "incident" that occurred the other week. I just want you to know I'm not some crazy wacko or anything, and if you got that impression of me then ... well, I just wouldn't know what to say. I'm totally not that way. Like, at all. Being well-adjusted and normal and super cool is basically what I do, so I'm mortified at the thought that you might not have gotten that vibe from me when I first came by your house.

Thing is, I had no money for a flight, so I hitchhiked to North Carolina in empty train cars and the beds of pickup trucks, which wasn't good for the homemade Marcus Fenix costume I was going to try to surprise you with. By the time I'd worn it all the way across the country, it fell apart a little bit, especially that boombox-looking chestplate thing. I apologize for treating the sacred COG armor in such a disrespectful way! I will fall on my chainsaw now, sir! Ha ha!

When I got to your address, I didn't see your primary car parked in the driveway—I checked the license plate and everything—so I thought I'd take a little peek inside, because I knew your house would be totally cool. Also, I thought maybe I could surprise you later on by mentioning how your kitchen would make a good deathmatch space, or how the master bedroom could use a little more texture density. I think I might have been trying to lean forward a little too much to see inside, though, so when I slipped forward and broke the glass, I totally fell inside the house, as if I was trying to break in! I was shocked! It figures that such a rich and famous guy like you would have a state-of-the-art home security system, but even I was surprised how fast the cops showed up.

I told the police you were a good buddy of mine, so I guess that's why they called you to the station. I know you hadn't ever seen me in real life before, but I wasn't lying when I said we've spoken—I once posted a reply on a message board that you posted on, like in the same thread and everything. I guess it was too much to ask you to remember that. I know you're very busy all the time and meet a lot of people and so on. Still, I thought maybe you'd recognize my costume, even after the officers crushed a lot of the remaining cardboard when they arrested me.

Please listen, Cliffy. I love you, man. I love everything about you. But I can't deny that I was hurt when you acted so coldly toward me then. It was like you thought I was a criminal. You didn't even crack a smile when I started singing Mad World at you. You might have broken my heart at that moment ...

Just kidding! Forgive and forget is what I say. That's all water under the bridge to me now. Once I get this felony charge taken care of and you hire me, I won't give you a hard time for that, or anything else. Nah. We'll have a good, hearty laugh about this misunderstanding, and then we'll make games together, forever and ever.

Sincerely,
Your stalker (Ha ha! Not really.)

MATTHEW WASTELAND *writes about games and game development at his blog, Magical Wasteland (www.magicalwasteland.com). Email him at* ***mwasteland@gdmag.com****.*

FANDEMONIUM

NOVEMBER 11-13, 2010 BATON ROUGE

RED STICK

INTERNATIONAL ANIMATION FESTIVAL

Is proud to introduce FANDEMONIUM, a complete Comic and Pop Culture Con merging the worlds of animation, art, video games, movies, and more!
Tables are at a discount through 7/15/10.
Red Stick's amazing lineup of speakers and events is still amazing!
(There are a few new surprises too!)

Cartoon-a-Palooza, Speakers and Workshops, Red Stick PITCH! competition, Princess Ball, The Red Stick Animation Awards, Screenings of Oscar nominated films, and much, much more.

To see our exciting 2010 festival lineup, visit www.redstickfestival.org.

LSU
CENTER FOR COMPUTATION & TECHNOLOGY