

gamedeveloper PRESENTS

# GAME CAREER GUIDE



FALL 2008

Student Cover (Emblem) Art  
Winner!



*Student Preferred!*

THE ANNUAL CLASSIC

*Everything You Never Knew  
About Making Games*

**SPECIAL ISSUE**  
**STARTING YOUR CAREER IN**  
**VIDEO GAMES**

» **TOP 10 TIPS**

FOR PROGRAMMERS, DESIGNERS, ARTISTS, AND MORE

» **STUDENT POSTMORTEM**

GESUNDHEIT! AN ART-HOUSE BOOGER GAME

» **QUIZ!**

GAME DEVELOPMENT  
IN 20 QUESTIONS

» **THE INTERVIEW**

HARD QUESTIONS AND  
HOW TO ANSWER THEM



36

## STUDENT POSTMORTEM

### 36 GESUNDHEIT!

Gesundheit! is an unusual game about a pig monster that eats boogers—but what's even more unusual is that its creator Matt Hammill made the game simply on his own initiative. He wasn't at a game school, and his university didn't have a game program, but where there's a will there's a way! This article discusses the highs and lows of his development experience.

By Matt Hammill

## DEPARTMENTS

### 2 GAME PLAN *By Jill Duffy*

Games in Abstraction

[LETTER FROM THE EDITOR]

### 5 WHO TO KNOW & WHAT TO DO

A primer to some of the industry's most important events and organizations.

[GAME DEV 101]

### 11 PAYCHECK PLEASE!

Information on salaries for entry-level developers. This could be you!

[GAME DEV 101]

### 27 WHAT IS A GAME ENGINE? *By Jeff Ward*

A high-level discussion of game engines and associated terms.

[GAME DEV 101]

COVER ART: JAMES CROW, GIN GRAVETTE, GREG WRIGHT

## FEATURES

### 19 HOW A GAME GETS MADE

Ever wonder about the process of making a game? This article walks you through the steps developers and publishers take to bring a traditional console game to market, beginning with that certain gleam in a developer's eye.

By Brandon Van Slyke

### 31 GAME JOB INTERVIEW QUESTIONS

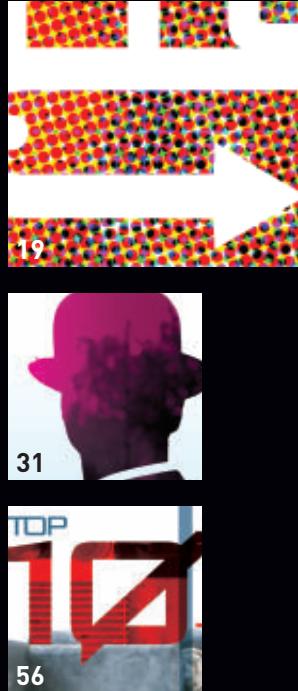
Nothing is scarier than interviewing for a job you actually want. Here, you'll see some common game job questions, and some guidance on how to answer them—and also how not to answer them.

By Jake Simpson

### 56 TOP 10 TIPS

Whether you're looking to enter the field of game design, production, programming, audio design, or art, we've got an expert at hand to give you a head start. These top 10 tips should not only help you focus your school time, but also improve your effectiveness once you do get that first job.

By Tim Lang, Michael J. Boccieri, Noel Llopis, Rob Bridgett, and Robert Chang



### 47 THE DISCIPLINES *By Albert T. Ferrier*

[GAME DEV 101]

Ever wonder what the people with these fancy job titles like game designer and programmer actually do? Wonder no longer!

### 51 STUDENT SURVEY *By Jill Duffy*

[EDUCATION]

What schools are prospective developers attending? What do they like about their classes? What don't they like? Massive stats lie within!

### 76 SCHOOL DIRECTORY

[EDUCATION]

A list of North American schools with game development classes, and a selection of international schools as well.

### 103 QUIZ *By Thomas Grové*

[GAME DEV 101]

How well do you know game history and game design?

### 108 ARRESTED DEVELOPMENT *By Matthew Wasteland*

[HUMOR]

Five ways to get into the game industry.

# GAME PLAN



## GAMES IN ABSTRACTION

### I'M NOT SURE ANYONE IN THE VIDEO GAME

industry knows where games are going. Massively multiplayer online games are hot—right?—as evidenced by WORLD OF WARCRAFT. Open-world games, like GRAND THEFT AUTO and all the lookalike games that followed thereafter, are hot—right? No, wait, Facebook application games are really where it's at. Episodic games are the new thing. Games for the iPhone are the new frontier.

No one in the game industry knows where games will be even two or three years from now. The only thing they know is that everything is changing and that the games that are released in a few years will be different from what we have now.

Who is going to drive that change?  
You are.

### NAME CHANGE

I was thinking the other day back to 2004 when Nintendo first gave word of its new console, code named Revolution. My game colleagues and I all thought Revolution was a stellar name. Rather than call it GameCube 2.0 or some other sequenced thing, Nintendo was promising grassroots change to the way we think about and play games. It was invigorating.

Then, in 2006, the official product name was announced: Wii. The press had a field day. We made ourselves giddy brainstorming headlines that played on all the meanings of the homonym: "diminutive," the exclamation (wee!), the anatomical meaning if you saw the word twice in a row, and of course, the verb.

Now, it's hard to imagine the whimsical Wii being called "Revolution," which in hindsight sounds dated, a thing of the late '60s.

Who knew the Wii would be what it has become? Who knew there would be an enormous software hole, desperate to be filled with new video games for the new way people play?

### THE NEW YOU

As the next generation of game developers, it's your job to shape what the game-playing public plays as well as how they play. In that sense, now is a truly exciting time to become a game developer.

I know many of you have dreams of joining top-name companies, like BioWare, Rockstar, Electronic Arts, and Ubisoft, but I hope others are raring to start their own companies. I hope some of you envision having your own studio, where you can push forward and make the new games that will be unique to your generation.

Whether you start your own studio or join the ranks of a well-funded corporation, it's important that you follow a path that suits your personality and lifestyle. That's true of your educational path as well. No job is perfect, and no school is without its bureaucratic moments. But finding the niche where you belong is important not only to your happiness, but also to your creativity.

As I've come to know different game developers, I've noticed distinct patterns about the personality types spread among the disciplines. [Need a primer on what I mean by "disciplines?"] Check out page 47.] As you read through this magazine, and as you encounter more game developers, try to think about the kinds of people who share the same mindset as you.

### WHAT CAME BEFORE

In art, the most innovative pieces—the artwork that inspired entire movements, like Cubism, Impressionism, and Dadaism—came from artists who learned the classical, foundational knowledge before they broke the rules. And so it is in games. Before you set out to change the world [which I know you will do], you need to know what's already there.

For starters, you'll need to know the complete process of how a video game is made. Brandon van Slyke, a game designer at Vicarious Visions, can walk you through all the steps starting on page 19.

Another thing you'll want to know is the common problems that come up in game development, as knowing what they are will quickly help you avoid them. The best way to learn that is by reading postmortems of finished games, and there's one waiting for you on page 36, by a student game developer, Matt Hammill.

### WHO'S YOUR MENTOR?

The great trendsetters in art—as well as science, philosophy, literature, and other fields—also had great teachers. I'm a huge advocate of mentorship in the game industry, and the only way you'll meet a mentor is if you get out and attend game industry functions, or go to a school that has industry veterans on staff or strong ties to developers in the community. Socialize. Learn. Think. Make great changes.

Make great games.

*Jill Duffy is senior contributing editor of Game Developer and editor of GameCareerGuide.com, a web site dedicated to helping aspiring game developers break into and learn about the industry.*

# game developer

Think Services, 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](mailto:gamedeveloper@halldata.com)

### EDITORIAL

**PUBLISHER** Simon Carless [scarless@gdmag.com](mailto:scarless@gdmag.com)

**SENIOR EDITOR** Brandon Sheffield [bsheffield@gdmag.com](mailto:bsheffield@gdmag.com)

**SENIOR CONTRIBUTING EDITOR** Jill Duffy [jduffy@gdmag.com](mailto:jduffy@gdmag.com)

**ART DIRECTOR** Cliff Scorsano [cscorsano@gdmag.com](mailto:cscorsano@gdmag.com)

**PRODUCTION EDITOR** Jeffrey Fleming [jfleming@gdmag.com](mailto:jfleming@gdmag.com)

**CONTRIBUTING EDITORS**

Jesse Harlin [jharlin@gdmag.com](mailto:jharlin@gdmag.com)

Steve Theodore [stheodore@gdmag.com](mailto:stheodore@gdmag.com)

Noel Llopis [nllopis@gdmag.com](mailto:nllopis@gdmag.com)

Soren Johnson [sjohnson@gdmag.com](mailto:sjohnson@gdmag.com)

Damion Schubert [dschubert@gdmag.com](mailto:dschubert@gdmag.com)

### ADVISORY BOARD

Hal Barwood Designer-at-Large

Ellen Guon Beeman Microsoft

Brad Bulkley Neversoft

Clinton Keith High Moon Studios

Ryan Lesser Harmonix

Mark DeLoura Ubisoft

### ADVERTISING SALES

#### MEDIA ACCOUNT MANAGER

John Malik Watson [jmwatson@think-services.com](mailto:jmwatson@think-services.com) t: 415.947.6224

#### GLOBAL SALES MANAGER, RECRUITMENT & EDUCATION

Aaron Murawski [amurawski@think-services.com](mailto:amurawski@think-services.com)

t: 415.947.6227

#### ACCOUNT MANAGER, EDUCATION AND RECRUITMENT

Gina Gross [ggross@think-services.com](mailto:ggross@think-services.com) t: 415.947.6241

#### SR. EVENTS ACCOUNT MANAGER, SOUTHWEST

Jasmin Davé

#### ACCOUNT MANAGER, WESTERN CANADA, INDIA, AUSTRALIA, & ASIA

Amanda Mae Miller

### ADVERTISING PRODUCTION

#### PRODUCTION MANAGER

Pete C. Scibilia [pscibili@ubm-us.com](mailto:pscibili@ubm-us.com)

### REPRINTS

#### PARS INTERNATIONAL

Joe Nunziata t: 212.221.9595 e: [reprints@parsintl.com](mailto:reprints@parsintl.com)

### THINK SERVICES

#### CEO THINK SERVICES

Philip Chapnick

#### GROUP DIRECTOR

Kathy Schoback

### AUDIENCE DEVELOPMENT

#### GROUP DIRECTOR

Kathy Henry [khenry@techinsights.com](mailto:khenry@techinsights.com)

#### DIRECTOR

Kristi Cunningham [kcunningham@techinsights.com](mailto:kcunningham@techinsights.com)

#### LIST RENTAL

Merit Direct LLC t: 914.368.1000

### MARKETING

#### SERVICES MARKETING DIRECTOR

Karen Tom [ktom@think-services.com](mailto:ktom@think-services.com)

#### SERVICES MARKETING COORDINATOR

Laura Robison [irobison@think-services.com](mailto:irobison@think-services.com)

### UBM TECHNOLOGY MANAGEMENT

#### CHIEF EXECUTIVE OFFICER

David Levin

#### CHIEF OPERATING OFFICER

Scott Mozarsky

#### CHIEF FINANCIAL OFFICER

David Wein

#### CHIEF INFORMATION OFFICER

Kevin Prinz

#### CORPORATE SENIOR VP SALES

Anne Marie Miller

#### SENIOR VP, STRATEGIC DEV. AND BUSINESS ADMIN.

Pat Nohilly

#### SENIOR VP, MANUFACTURING

Marie Myers

#### SENIOR VP, COMMUNICATIONS

Alexandra Raine

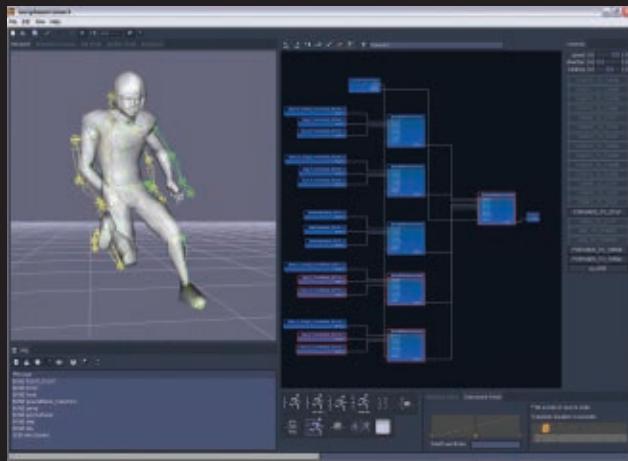


United Business Media

WWW.CMPGAME.COM

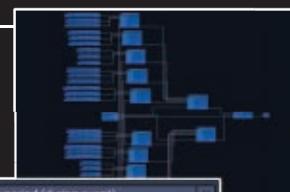
# morpheme™

advanced animation system



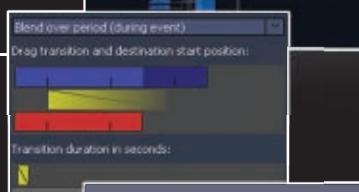
## blend tree

Advanced graphical tools for building complex blend trees. Real-time visualization of animation source contribution through node highlighting



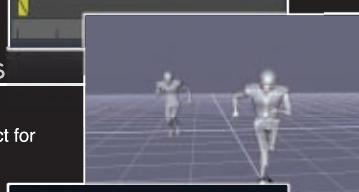
## blending

Graphical control of transition blending between states in the transition graph



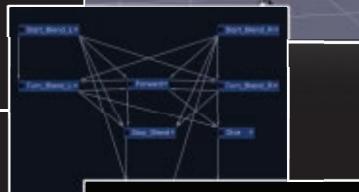
## multiple characters

Visualization of multiple runtime characters in morpheme:connect for easy authoring and analysis of character interaction



## network

Advanced graphical tools for creating and visualizing transition networks through drag-and-drop



## control parameters

Exposure of custom high-level controls for entire animation system. Real-time manipulation through sliders or game pad controller

speed 0.0	1.0
direction 0.0	1.0
banking 0.0	1.0

morpheme is the industry's first graphically authorable animation engine. morpheme consists of morpheme:runtime: an advanced runtime animation engine for PLAYSTATION®3, Xbox 360™, Wii™ and PC. morpheme:connect: a highly-customizable 3D authoring application.

morpheme gives animators and developers unprecedented control over the look and feel of their animations in-game: blends, transitions, compression, etc. can all be previewed and modified graphically in morpheme:connect and live on the target platform.

morpheme:runtime ships with full source code and integrates seamlessly with euphoria, NaturalMotion's Dynamic Motion Synthesis technology.

For more information, visit [www.naturalmotion.com](http://www.naturalmotion.com)

## scripting

Full Lua scripting for automating tasks, adding AI logic or polling game pads for real-time input

## timeline

Graphical mark up of animation data to add one-shot and duration events, for highlighting footfalls, sound effects, etc.

## node palette

Advanced blend nodes for dragging and dropping into transition network. Fully customizable node types through C++ and scripting

## animation browser

Easy browsing and selection (drag & drop) of source animation. Animation list is automatically updated to reflect changed source files

## transition requests

Exposure of custom transition messages. In-tool emulation of interaction between morpheme:runtime and game AI system



# WHO TO KNOW & WHAT TO DO IN THE GAME INDUSTRY

## EVENTS

One of the best ways to become better informed about the video game industry is to attend its major events—here's a primer.

### GAME DEVELOPERS CONFERENCE

[www.gdconf.com](http://www.gdconf.com)

The yearly Game Developers Conference has grown from its origins as an informal living room meeting to become a weeklong event attended by thousands. The conference has a long list of speakers each year and the offerings are scheduled along tracks. Lectures and summits are grouped into Audio, Business & Management, Game Design, Production, Programming, Visual Arts, and many other categories so that attendees can experience a full range of viewpoints within their chosen discipline. The GDC also hosts the Independent Games Festival and Game Developers Choice Awards. There is a large exhibition hall where companies can showcase their development tools, and students will want to explore the GDC's Career Pavilion where they can network with development studios who are looking for fresh talent. GDC nights are given over to a dizzying array of parties, dinners, and pub-crawls. GDC is owned and operated by Think Services, which also owns Game Developer and Gamasutra.com. The next GDC will be held March 23-27, 2009 in San Francisco, California.

### AUSTIN GAME DEVELOPERS CONFERENCE

[www.austingdc.net](http://www.austingdc.net)

Austin is quickly gaining recognition as one of the prime gathering places for development talent in the Southwest. Meeting the needs of this growing community, the Austin Game Developers Conference provides three days of workshops, speakers, and summits for both professionals and students. The Austin GDC has a special focus on game audio, game writing, and online gaming and like its bigger sibling in San

Francisco it hosts the WorldsInMotion Summit, the Independent Games Festival Showcase, and a Game Career Seminar. Organized by Think Services, the Austin GDC takes place September 15-17, 2008 in Austin, Texas.

### GAME CONVENTION

[www.gc-germany.de](http://www.gc-germany.de)

One of Europe's major video game trade shows, the Game Convention is held each summer in Leipzig, Germany. The convention, which is open to the public, gives exhibitors an opportunity to connect with their audience on a vast show floor. Game creators can also participate in the GC Developers Conference that features lectures on all aspects of game development.

### E3 MEDIA & BUSINESS SUMMIT

[www.e3summit08.com](http://www.e3summit08.com)

This event was previously called the Electronic Entertainment Expo and it was the yearly Woodstock of video games. The event, which is run by the ESA, was originally created as an industry-only trade show. However, as video games became mainstream entertainment, E3 began to attract tens of thousands of attendees, many of whom had only a tenuous connection to the industry. Publishers responded to the crowds with increasingly elaborate demo booths each year and the costs of putting on the extravagant show skyrocketed. Beginning with the 2007 event, E3 became the E3 Media & Business Summit and was refocused as a much smaller and more sober affair that would



The Game Developers Conference 2008.

PHOTOGRAPH BY VINCENT DIAMANTE

PHOTOGRAPH BY BRANDON SHEFFIELD



The seamy underbelly of the Tokyo Game Show 2007.

be strictly invitation-only. Taking place in July in Los Angeles, the summit brings together a small group of major retailers, publishers, and media for meetings and demonstrations without the floor show overkill.

In response to the ESA's consolidation of E3, a separate event (not affiliated with the ESA) was organized called the E for All Expo. E for All takes place in October in Los Angeles and gives the public an opportunity to demo many of the newest games that are coming down the retail pipeline.

### CHINAJOY

<http://en.chinajoy.net>  
ChinaJoy, the China Digital Entertainment Expo and Conference, is a yearly event that brings together the Chinese game industry with publishers and developers across the globe. Open to both the trade and the public, ChinaJoy combines business conferences with a floor show exhibition for consumers. The event is held each July in Shanghai.

### PENNY ARCADE

[www.pennyarcadeexpo.com](http://www.pennyarcadeexpo.com)  
The Penny Arcade gaming convention is a three-day, end-of-summer weekend filled with a deep and abiding love for video games. Thousands of attendees meet at the expo to play new and upcoming games, spend time with old favorites, and compete in the epic Omegathon elimination tournament. There is also an exhibition hall for game companies to show off their latest wares and a roster of speakers to take in as well. The expo also features the PAX 10, a showcase for outstanding independent games. Penny Arcade Expo will take place this year in Seattle, August 29-31.

### D.I.C.E. SUMMIT

[www.dicesummit.org/index.php](http://www.dicesummit.org/index.php)  
Since 2002, the Academy of Interactive Arts & Sciences has been holding the D.I.C.E. (Design, Innovate, Communicate, Entertain) Summit each year in Las Vegas. The summit brings together big money people from publishing and development to discuss game industry trends and opportunities. While much of the summit is given over to high-level business talks, the AIAS also takes the occasion to present its annual best of Interactive Achievement Awards.

### TOKYO GAME SHOW

<http://tgs.cesa.or.jp/english/>  
The Tokyo Game Show is the largest annual video game trade-show in Japan and it fires the starting guns of the Christmas buying season with four days of new game previews and announcements. While the first two days of the show are restricted to industry people only, the last two days are open to the public. In addition to its enormous exhibition floor, TGS hosts workshops and conferences, as well as the Computer Entertainment Supplier's Association's (CESA) annual Japan Game Awards. TGS is co-organized by CESA and Nikkei Business Publications, Inc. and will be held this year on October 9-12 at the Makuhari-Messe in Chiba City.

## ORGANIZATIONS

These associations help advocate, regulate, and advance the industry. Here are the big names you should know.

### IGDA

[www.igda.org](http://www.igda.org)

The International Game Developers Association (IGDA) is a non-profit organization that works to connect and educate game developers at all levels in the industry. The IGDA has local chapters across North America, Asia, and Europe where members can meet with their peers and the organization sponsors events and parties at the many developer conferences that take place each year. The IGDA is also a rich online information source for developers, with a website full

of specialized Wikis, SIGs, White Papers, articles, and columns that address a wide variety of game development issues. Access to the website is free and annual memberships are \$30 for students and \$48 for professionals.

### ESA

[www.theesa.com](http://www.theesa.com)

The Entertainment Software Association (ESA) represents the interests of U.S. video game publishers. The ESA supports the industry with anti-piracy programs, government outreach, market research,

and intellectual property protection initiatives. The ESA runs the E3 Media & Business Summit and supports a number of charities through its ESA Foundation.

### ESRB

[www.esrb.org](http://www.esrb.org)

The Entertainment Software Rating Board (ESRB) was created by the ESA to rate the content of video games released in North America and to ensure that the industry follows a common set of advertising guidelines. The ESRB enjoys broad support from both publishers and retailers.



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 Front Line Award for Best Engine for the past three years, and Epic was awarded Best Studio at the 2006 Spike TV Video Game Awards. Epic's "Gears of War," the 2006 Game of the Year, has sold over 4.7 million units on Xbox 360 and PC. Epic recently shipped "Unreal Tournament 3" for PC and PlayStation 3, and is on track to ship the Xbox 360 version this summer. "Gears of War 2" for Xbox 360 is scheduled for release in November.

#### **Upcoming Epic Attended Events:**

**E3 2008**  
Los Angeles, CA  
July 15-17, 2008

**Microsoft Gamefest**  
Seattle, WA  
July 22-23, 2008

**Casual Connect**  
Seattle, WA  
July 24-25, 2008

**GC Developers Conference**  
Leipzig, Germany  
August 18-20, 2008

Please email:  
[mrein@epicgames.com](mailto:mrein@epicgames.com)  
for appointments.



# Unreal Technology News

by Mark Rein, Epic Games, Inc.

### **EPIC BUYS CHAIR ENTERTAINMENT, CREATOR OF XBLA GAME OF THE YEAR *UNDERTOW***

Epic Games has acquired Chair Entertainment, the Provo, Utah-based studio behind the underwater shooter *Undertow*, Official Xbox Magazine's 2007 Xbox Live Arcade Game of the Year, which is powered by Unreal Engine 3.

While Chair's team is small in size, its agility, discipline and strategic planning have enabled the studio to efficiently produce high-quality, downloadable games using the Unreal Engine; for example, *Undertow* shipped with a footprint of only 48 MB.

Epic plans to leverage Chair's expertise to better serve the downloadable and casual game market going forward.

Chair has been developing an original, cross-media property called *Empire*, a saga about a second, near-future American Civil War.

Early in the project, co-founders Donald and Jeremy Mustard enlisted the help of renowned author Orson Scott Card, with whom they and other Chair core team members had previously collaborated on *Advent Rising* for Xbox and PC.

*Empire* began as a game with an in-house written story, and in 2006, Card expanded the narrative into a *New York Times* best-selling novel. Warner Bros. and producer Joel Silver have optioned the film rights.

Chair also owns the license to create the video game adaptation of Card's classic 1985 science fiction novel, *Ender's Game*.

### **AVALON STYLE LICENSES UNREAL ENGINE 3 FOR OFF-ROAD DRIVING GAME**

Avalon Style Entertainment, one of Russia's leading developers of interactive entertainment, has licensed Unreal Engine 3 to develop the latest installment in its popular "4x4 Off Road" off-road driving simulation series for PC, Xbox 360™ and PLAYSTATION®3.

"4x4 Off Road III" (international title, "Off Road Drive"), will be published by 1C Company.

Avalon Style is one of the first Russian game studios to license Unreal Engine 3 for cross-platform computer and video game development.

While previous releases in its "4x4 Off Road" racing franchise utilized the company's proprietary racing technology, the development team was convinced to license Unreal Engine 3 after first-hand experience with its comprehensive infrastructure and tools.

"Unreal Engine 3 is at the forefront of multiplatform game development," said Vyacheslav Gordeyev, CEO of Avalon Style.

"The technology has been successfully used in many critically acclaimed titles, and we feel fully armed to make the first off-road driving simulator on Unreal Engine 3 thanks to built-in vehicle support, a powerful PhysX system, and many other leading edge features."



Jeremy and Donald Mustard are the co-founders of Chair Entertainment, a wholly-owned subsidiary of Epic.

### **SEGA LICENSES UNREAL ENGINE 3 FOR NEW CROSS-PLATFORM TITLE**

SEGA® of America has extended its relationship with Epic by signing a new Unreal Engine 3 licensing deal for an unannounced, cross-platform game.

SEGA entered into its original licensing agreement to create multiple titles with Unreal Engine 3 in May 2007.

"The great capabilities of Unreal Engine 3 make it possible for our developers to deliver incredible gaming experiences," said Dave Cobb, Vice President of Development, SEGA of America, Inc.

Cobb continued, "Unreal Engine 3 provides superior power and flexibility enabling SEGA to bring original and exciting next-generation, multi-platform titles to market."



For UE3 licensing inquiries email:  
[licensing@epicgames.com](mailto:licensing@epicgames.com)

For Epic job information visit:  
[www.epicgames.com/epic\\_jobs.html](http://www.epicgames.com/epic_jobs.html)

WWW.EPICGAMES.COM

**VIDEO GAMES VOTER NETWORK****[www.videogamevoters.org](http://www.videogamevoters.org)**

The Video Games Voter Network is a web site created by the ESA to rally support against government regulation of video games. It encourages video game consumers to involve themselves in the democratic process by providing voter registration information and it organizes online petitions to stop legislative efforts to restrict game sales.

**ELSPA****[www.elspa.com](http://www.elspa.com)**

The Entertainment & Leisure Software Publishers Association (ELSPA) is the voice of the British game industry, promoting its interests in both the business and government sectors as well as at the consumer level. ELSA tracks game sales, conducts market research, and maintains a strong anti-piracy initiative. The organization also works to educate consumers on game ratings and is a sponsor of the London Games Festival.

**TIGA****[www.tiga.org](http://www.tiga.org)**

The Independent Games Developers

Trade Association (TIGA) is a U.K.-based trade group that promotes European game development interests. TIGA is an advocate for game industry concerns and works with government departments and ministers to improve business opportunities within the U.K. and across the European Union. The organization also functions as a networking resource for the industry and maintains a best practices handbook that collects information on contracts, tax credits, grants, insurance, outsourcing, and PEGI (Pan European Game Information) age ratings.

**AIAS****[www.interactive.org](http://www.interactive.org)**

The Academy of Interactive Arts & Sciences (AIAS) is a non-profit organization open to industry professionals that promotes and recognizes developer excellence. Each year the AIAS presents its Interactive Achievement Awards that honor the best of the game industry's work. The Academy also organizes the annual D.I.C.E. Summit that brings together industry leaders for high-level seminars

and discussions. Dues-paying membership with full voting rights is open to individuals from the creative and technical fields that have been active in the game industry for at least two years while membership with limited voting rights is open to those in the business sector and to students who plan on joining the industry.

**CESA****[www.cesa.or.jp/index.php/en](http://www.cesa.or.jp/index.php/en)**

The Computer Entertainment Supplier's Association (CESA) is a Japanese trade group that promotes the video game industry through events like the Tokyo Game Show and the CESA Developers Conference. It also oversees the Computer Entertainment Rating Organization, which is roughly equivalent to our own ESRB. CESA organizes several committees that address a variety of industry concerns, from technology and intellectual property to market research, distribution, and human resources. ☀

## additional resources

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

**EVENTS**

- MI6 (advertising in games)
- Casual Connect Seattle (and other worldwide casual gaming shows)
- Microsoft Gamefest
- Montreal Game Summit
- Siggraph (major symposium for graphic arts industry)
- Taipei Games Show
- Game Connection (Multiple Events)
- GDC Mobile (held during GDC)
- Vancouver International Game Summit

**ORGANIZATIONS**

- Computer Game Artists Association
- Game Developers Association of Australia
- International Game Journalists Association
- G.A.N.G. (for developers of game audio and music)
- IGDA Student Action special interest group
- NPD (market research firm that reports on games)
- The Entertainment Consumers Association (ECA)

**WEB RESOURCES**

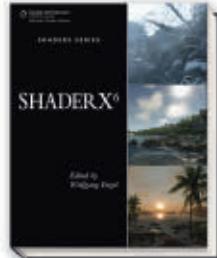
- GameCareerGuide.com
- Gamasutra.com
- GamesIndustry.biz
- Next-Gen.biz
- Gamedev.net
- GameSpot.com
- GameJobs.com
- Gamedevmap.com
- Gamepolitics.com
- Gamestudies.org (site for The International Journal of Computer Game Research)

# YOUR ULTIMATE RESOURCE

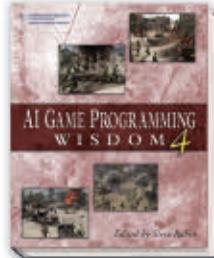
Course Technology PTR's game development list offers coverage of everything from behind-the-scenes programming and awesome character design, to storyline development and special effect creation, using the latest game creation toolkits, software, and engines. We also offer titles that teach essential advanced-level skills such as math and physics, and unique topics including business and marketing and legal issues.



**Beginning Scripting  
Through Game Creation**  
1-59863-511-5 • \$29.99



**ShaderX6  
Advanced Rendering  
Techniques**  
1-58450-544-3 • \$59.99



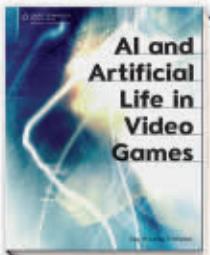
**AI Game Programming  
Wisdom 4**  
1-58450-523-0 • \$69.99



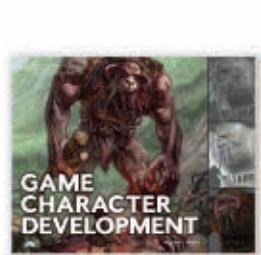
**Best of Game  
Programming Gems**  
1-58450-571-0 • \$49.99



**The Official  
Luxology modo Guide  
Version 301**  
1-59863-497-6 • \$49.99



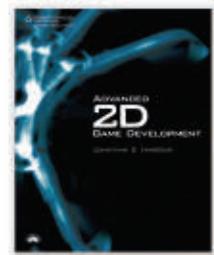
**AI and Artificial Life  
in Video Games**  
1-58450-558-3 • \$39.99



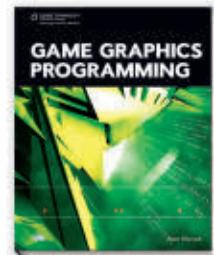
**Game Character  
Development**  
1-59863-465-8 • \$44.99



**Maya Feature  
Creature Creations  
Second Edition**  
1-58450-547-8 • \$49.99



**Advanced 2D  
Game Development**  
1-59863-342-2 • \$59.99



**Game Graphics  
Programming**  
1-58450-516-8 • \$64.99

---

**Challenges  
for Game Designers**  
1-58450-580-X • \$24.99  
August 2008

**Ultimate Game  
Programming with DirectX  
Second Edition**  
1-58450-559-1 • \$49.99  
September 2008

**Collaborative Online  
Game Creation**  
1-58450-560-5 • \$39.99  
October 2008

**AI Game Engine  
Programming  
Second Edition**  
1-58450-572-9 • \$54.99  
November 2008

**The Complete XNA  
An Exploration into the  
XNA Framework Library**  
1-58450-537-0 • \$49.99  
November 2008

Visit us in **Booth #1019** at **SIGGRAPH 2008**. Stop by for special show-only discounts on our comprehensive list of animation, graphics, and game development titles.

Not coming to SIGGRAPH 2008?

Our complete line of books are available at Amazon, Barnes & Noble, Borders, and other fine retailers nationwide, online at [www.courseptr.com](http://www.courseptr.com), and by phone at **1.800.354.9706**.



**COURSE TECHNOLOGY**  
CENGAGE Learning™  
Professional • Technical • Reference

**courseptr.com**



jill duffy



# PAYCHECK PLEASE!

## AVERAGE SALARIES OF ENTRY-LEVEL GAME DEVELOPERS

### » "HOW MUCH MONEY WILL I MAKE AS A GAME DEVELOPER?"

You asked. We listened. Then we asked literally thousands of game developers to tell us how much they make in a year [see the Methodology on page 16 for full details]. We gathered those hard numbers and hashed them out, configuring average salaries for video game programmers, designers, artists, and more.

Very few game-makers strike it rich. Not many pull six-figure salaries.

Although the job of creating video games sounds glamorous to many outsiders, the insiders will tell you they earn an honest living and work much harder than what their pay should demand of them.

But—and here's the perk of all perks—you get to do what you love.

### NEGOTIATION PREPARATION

First-time job applicants in the game industry need to know what kinds of numbers to expect when they start negotiating a job offer, and this survey can prepare them well for that. The key is to identify the job title, region where the job is located (see page 16), and years' experience—if you have that much in hand, this survey can go a long way toward helping you create realistic expectations.

If you are female and have been offered a job in game development, ask for more money before you accept. Women are routinely

underpaid in this industry (as is shown at the bottom of each page in this article), and the incoming generation of developers is the only one that has a shot at correcting it before their careers start. It's true that women are underpaid by about 90 cents on the dollar compared to men in nearly all lines of work, but games are a progressive business and can do better!

### STUDENT LOANS?

If you're a student or are looking to enter higher education, another reason it's important to have realistic salary expectations is to help you make financial decisions about how much tuition you're willing to pay and how much money in loans you're willing to borrow. Even though game development is an extremely competitive industry to get into, the financial reward is not commensurate—it's the job satisfaction that makes up for it. But if you aspire to a lifestyle like Chamillionaire's, you probably should think about majoring in economics, or taking your computer science degree into database programming for the financial sector and cut your losses now.

On the other hand, if you're content to take your \$35K to \$40K entry-level salary (a bit higher for game engineers, but not by much), pay off your loans on the standard 10-year plan, work harder than you ever have, and love your job like none other, then at least you know what you're getting into.

Jill Duffy is senior contributing editor of Game Developer and editor-in-chief of GameCareerGuide.com, a web site dedicated to helping aspiring game developers break into and learn about the industry. Email her at [jduffy@gdmag.com](mailto:jduffy@gdmag.com).

[WWW.GDMAG.COM](http://WWW.GDMAG.COM) 11

# PAYCHECK PLEASE!

## PROGRAMMERS

### PROGRAMMERS REMAIN THE MOST

in-demand employees of the game industry. Good ones are hard to find, and when they are found, they're compensated appropriately. Entry-level salaries in the U.S. start at around \$60K per year for programmers, though across all levels of experience, their average jumps to around \$83K per year.

For programmers with a bachelor's degree, some college, or an associate's degree, we found no clear correlation between salary and education. The factors that seem to have a greater affect than education on game programmers' salaries are: work experience, having both general and specialized knowledge, and talent. The exception is programmers with a master's degree, who earn significantly more than their peers, close to \$91K.

There is still a strikingly low representation of women in game programming (only 2.6%), and an average salary difference of more than \$7,000, in favor of males.

Avg. Salary for  
Three or Fewer  
Years' Experience

**\$60,296**

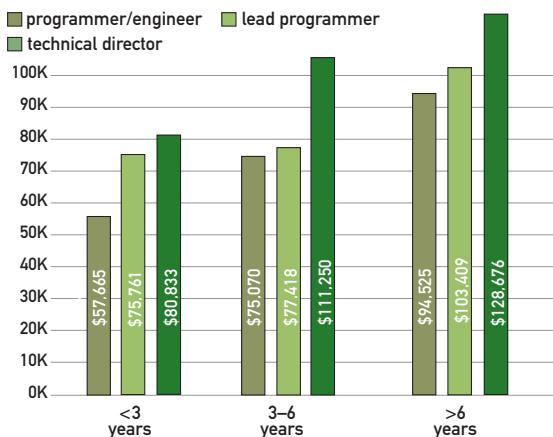
Avg. Salary  
Across All Levels  
of Experience

U.S. **\$83,383**

Canada **\$70,167**

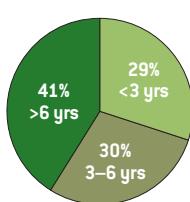
Europe **\$53,146**

### Programming salaries per years of experience and position



ALL PROGRAMMERS AND ENGINEERS

### Years experience in the industry



### Gender Stats for Programmers

Gender	Percent represented	Average salary
Female	3%	\$76,184
Male	97%	\$83,579

### Percent receiving additional compensation

78%  
Average additional compensation **\$15,313**

### Type of compensation

- Annual bonus ..... 50%
- Project bonus ..... 28%
- Royalty ..... 17%
- Stock Options ..... 41%
- Profit Sharing ..... 17%

### Receive some benefits

94%  
Type of benefits received

- Medical ..... 97%
- Dental ..... 92%
- 401K/retirement ..... 84%

## ARTISTS

### OVER THE PAST FEW YEARS, VIDEO GAME

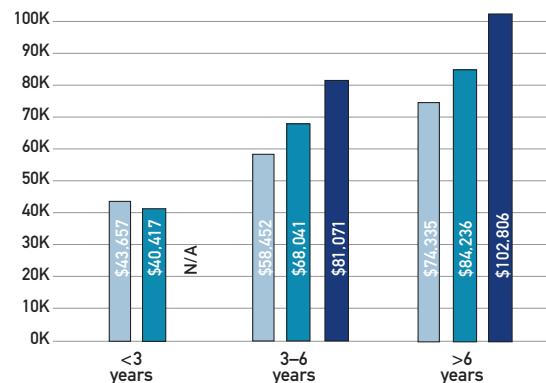
artists have had to come to grips with a dirty word: outsourcing. Art assets used in video games can easily be created and polished outside the main studio, so it's easy for game companies to send more and more art-based work overseas, where labor is cheaper. This doesn't mean that game artists' jobs are in short supply. What it does mean is that the desired skill sets of full-time in-house artists are changing.

To supervise all the outsourced work flowing into and out of the studio, game companies need art managers, people who speak the language of artists and have an artistically trained eye. Other jobs in the art department that need to stay in-house include concept artist, animator, and rigger.

I've also been hearing (anecdotally from professional game artists and human resources teams in the game industry) that artists who have some technical knowledge are better positioned as job candidates than those who have none. Artists whose resumes show they know a scripting language or have some general knowledge of programming tend to stand out in the eyes of the hiring committee.

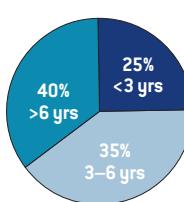
### Art and animation salaries per years of experience and position

artist & animator      lead artist      art director



ALL ARTISTS AND ANIMATORS

### Years experience in the industry



### Percent receiving additional compensation

80%  
Average additional compensation **\$14,984**

### Type of compensation

- Annual bonus ..... 44%
- Project bonus ..... 31%
- Royalty ..... 22%
- Stock Options ..... 37%
- Profit Sharing ..... 17%

### Gender Stats for Artists

Gender	Percent represented	Average salary
Female	8%	\$61,250
Male	92%	\$67,056

### Receive some benefits

91%  
Type of benefits received

- Medical ..... 99%
- Dental ..... 95%
- 401K/retirement ..... 87%

# ACTIVISION®

## GREAT GAMES — START WITH — GREAT PEOPLE



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



Visit our site: [activision.com](http://activision.com)

# PAYCHECK PLEASE!

## GAME DESIGN

**A FREQUENT TOPIC OF CONVERSATION FOR ME IN THE** past year has been about whether a fresh-out-of-college graduate can get his or her first game development job as a game designer. Can you break into the industry as a designer, or do you have to earn your stripes elsewhere before the design department will let you in?

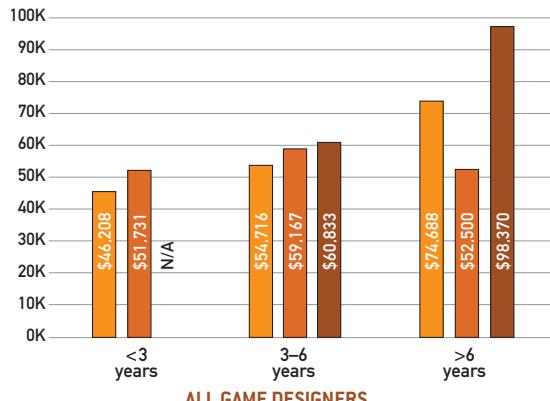
The answer is: Yes, your first job in game development can be as a game designer. However, the trick is in understanding the different kinds of design jobs that are available, and knowing which ones are better suited to entry-level candidates. [For a more complete explanation of the job titles within the design department, see "The Disciplines," page 47.]

Game designers can earn a comfortable living, but students and parents who are still learning about the profession should be clear that it is not generally conducive to a wealthy lifestyle. On average, game programmers make about \$20,000 more per year than designers (and game programmers earn far less than their peers in other fields, like banking or security). The need for game designers is also not as strong as the need for programmers, making the job market highly competitive.

To become a singer, one must sing. To become a writer, one must write. And to become a game designer, one must design games—before applying for the job. Students of game design should work on video game projects with artists, programmers, and producers. They can also exercise their game design skills by making non-electronic games, such as board games, card games, and puzzles.

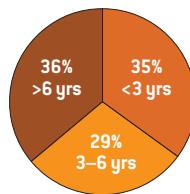
### Game design salaries per years of experience and position

■ game designer ■ writer ■ creative director/lead designer



ALL GAME DESIGNERS

### Years experience in the industry



### Gender Stats for Designers

Gender	Percent represented	Average salary
Female	8%	\$55,156
Male	92%	\$64,396

**Avg. Salary for three or fewer years' experience**

**\$46,184**

**Avg. Salary across all levels of experience**

**U.S. \$63,649**

**Canada \$57,435**

**Europe \$46,959**

## AUDIO

**Avg. Salary for three or fewer years' experience**

**\$53,971**

**Avg. Salary across all levels of experience**

**U.S. \$73,764**

**Canada \$56,750**

**Europe \$62,000**

**JOBS IN VIDEO GAME AUDIO ARE SCARCE.** FEW

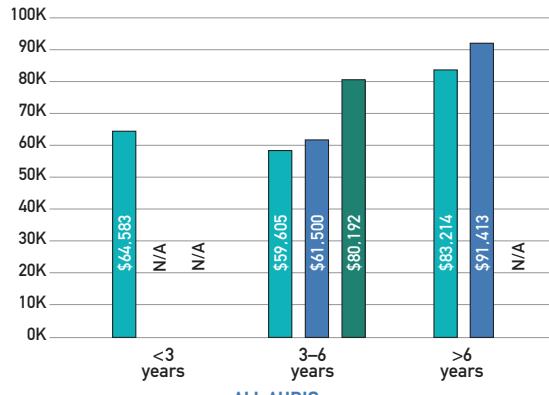
full-time in-house positions exist; much of the workforce is freelance or contract. Audio people are often in between worlds, and that niche area that they inhabit is a realm where only true expertise survives. They're educated in the world of audio engineering and sound design, but have a personal interest in video games. The sounds they record and create for video games all have extremely specific considerations and requirements: Will the track loop? What information does the sound give the player? How many variations of a particular sound effect are needed to keep the player from auditory boredom? Is the dialogue critical to the story or gameplay? And because audio is often implemented very late in a game's development: How little memory is there left to work with?

Audio people in the game industry reported earning almost \$19,000 per year in additional income, possibly because so many of them are freelancers, though this explanation is not grounded in any hard data.

### Audio salaries per years of experience and position

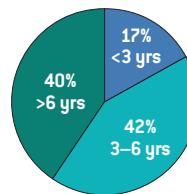
■ sound/audio designer or engineer ■ sound/audio director

■ composer/musician



ALL AUDIO

### Years experience in the industry



**Percent receiving additional compensation** 66%

**Average additional compensation** \$18,819

### Type of compensation

Annual bonus .....	38%
Project bonus .....	29%
Royalty .....	31%
Stock Options .....	26%
Profit Sharing .....	22%

### Gender Stats for Audio Personnel

Gender	Percent represented	Average salary
Female	8%	\$69,375
Male	92%	\$73,764

**Receive some benefits** 83%

**Type of benefits received**

Medical .....	97%
Dental .....	95%
401K/retirement .....	85%

## PRODUCTION

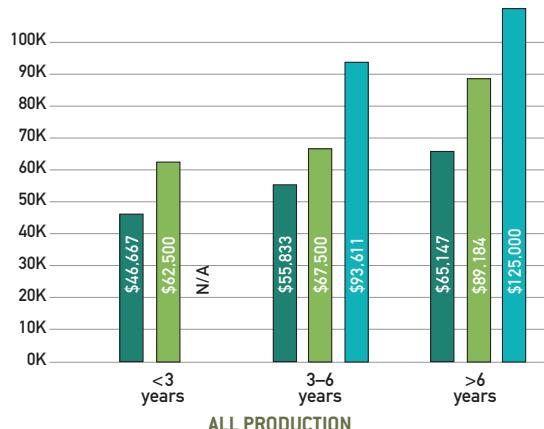
ENTRY-LEVEL PRODUCER JOBS HAVE SEEN AN up tick in the last two to three years—so much so that the editors of Game Developer [who administer the survey that this data is based on] were actually dogged by survey respondents to add the job title “assistant/associate producer.” Associates or assistants, which usually refer to the lesser experienced producers, earned on average \$55,140 per year, a figure that’s pretty close to the \$52,763 reported by all producers with 3 or fewer years experience.

What should an entry-level producer candidate highlight on his or her resume? The job is similar to project management, but also requires some knowledge of video games, so aspiring game producers should involve themselves in as many game-related projects and events as possible. When electronic game projects are scarce, volunteer work and community outreach projects can also afford similar opportunities. The role requires leadership skills, a mediator’s mentality, and organization to a critical degree. Producers need to be good listeners and should be able to get along with many types of people. They need face-to-face approachability just as much as they need strong written communication skills.

An interesting statistic about production is that there are more women in this department than any other in game development, with the exception of business [which includes administration, public relations, marketing, human resources, and executive roles].

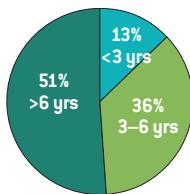
### Production salaries per years of experience and position

■ associate producer   ■ producer/project lead   ■ executive producer



ALL PRODUCTION

### Years experience in the industry



### Gender Stats for Producers

Gender	Percent represented	Average salary
Female	18%	\$72,398
Male	82%	\$79,970

AVG. SALARY FOR THREE OR FEWER YEARS' EXPERIENCE

**\$52,763**

AVG. SALARY ACROSS ALL LEVELS OF EXPERIENCE

U.S. **\$79,970**

CANADA **\$71,786**

EUROPE **\$59,792**

## QUALITY ASSURANCE

OF ALL THE DISCIPLINES, VIDEO GAME TESTERS

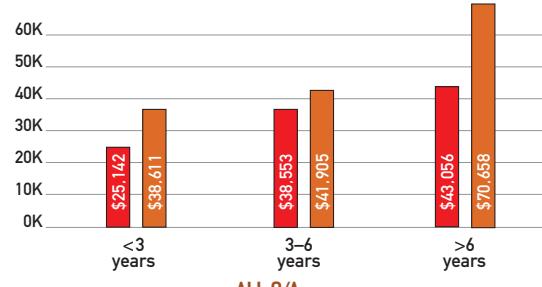
have the highest representation of entry-level people—44% of them have 3 or fewer years’ experience. Academically speaking, they tend to have some college, an associate’s degree, or a bachelor’s degree. So few testers held more advanced degrees that their numbers were considered insignificant for the purpose of the Salary Survey. Most students who are serious about making a career in game development view a role in QA as only a temporary position, a quick fix if they cannot land a job in the discipline of their choice. Taking a job as a tester can in fact be a stop-gap measure, a way to earn a pay check while also learning how a game development company functions from the inside. But occasionally, game developers find a good home among testers and stick around long enough to earn the more advanced “lead tester” job title. QA leads, across all number of years experience, earned on average \$48,070 per year, more than the least experienced artists and designers reported earning.

When the title “lead tester” was excluded and the average salary calculated across all levels of experience, QA testers made on average \$30,278 per year, very close to the \$28,556 per year earned by those with 3 or fewer years experience.

Finally, game testers in Europe and Canada do surprisingly well financially when measured against their U.S. counterparts [take a look at the differences in pay in other disciplines].

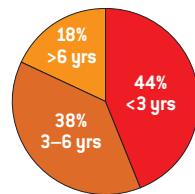
### Q/A salaries per years of experience and position

■ tester  
■ Q/A lead



ALL Q/A

### Years experience in the industry



### Receive some benefits

95%

Type of benefits received

Medical ..... 97%

Dental ..... 93%

401K/retirement ..... 86%

### Gender Stats for Testers

Gender	Percent represented	Average Salary
Female	6%	\$34,375
Male	94%	\$39,309

Percent receiving additional compensation

59%

Average additional compensation

\$8,833

Type of compensation

Annual bonus ..... 64%

Project bonus ..... 15%

Royalty ..... 2%

Stock Options ..... 37%

Profit Sharing ..... 6%

Receive some benefits

73%

Type of benefits received

Medical ..... 96%

Dental ..... 86%

401K/retirement ..... 79%

# PAYCHECK PLEASE!

## METHODOLOGY

**THE DATA AND COMMENTS IN THIS ARTICLE HAVE** been developed exclusively for *Game Career Guide* readers to focus on entry-level salaries and other statistics most relevant to newcomers to the game industry. The results are based on the findings of the 2008 "Game Developer Salary Survey," which looked at salaries for the year ended 2007 and was conducted by *Game Developer* magazine and research firm Audience Insights in February and March 2008. A complete survey was originally published in the April 2008 edition of *Game Developer*.

We gathered 4,863 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 under \$10,000 as well as salary figures from students and educators. The small number of reported salaries over \$202,500 were excluded to prevent their high numbers from unnaturally skewing the average. We also excluded records that were missing key demographic and classification numbers.

The survey primarily includes U.S. compensation but consolidated figures for Canada and Europe were included as well. The total sample reflected in the data presented for the U.S. is 2,409, for Canada 394, and for Europe 382.

The sample represented in our salary survey can be projected to the overall game developer community with a margin of error, for the aggregate U.S. statistics, of plus or minus 1.7% at a 95% confidence level. The margin of error increases for specific subgroups reported within this community.

## GEOGRAPHY MATTERS

**TOP TEN EARNERS BY U.S. STATE (ACROSS ALL DISCIPLINES AND LEVELS OF EXPERIENCE)**

RANK	STATE	AVG. SALARY
1	California	\$81,502
2	Washington	\$74,989
3	Illinois	\$74,864
4	Maryland	\$73,566
5	New York	\$69,747
6	Georgia	\$69,265
7	Oregon	\$69,118
8	Virginia	\$67,619
9	Massachusetts	\$67,500
10	Colorado	\$66,638

## CANADA

**TOP 4 CANADIAN PROVINCES BY REPRESENTATION (AVERAGE SALARY ACROSS ALL DISCIPLINES AND LEVELS OF EXPERIENCE)**

1. BRITISH COLUMBIA	\$71,227
2. QUEBEC	\$56,867
3. ONTARIO	\$59,615
4. ALBERTA	\$68,500

## EUROPE

**TOP 5 EUROPEAN COUNTRIES BY REPRESENTATION**

1. UNITED KINGDOM
2. FRANCE
3. GERMANY
4. SPAIN
5. SWEDEN *

\*The Nordic countries are often viewed collectively in game development. When combined, the total representation from Nordic countries places the region in the no. 2 position.

## THE GENDER GAP

**AVERAGE SALARIES BY DISCIPLINE AND GENDER (ACROSS ALL LEVELS OF EXPERIENCE)**

	Male	Average Salary	Female	Average Salary
<b>Programming</b>	97.4%	\$83,579	2.60%	\$76,184
<b>Art</b>	92.0%	\$67,056	8.0%	\$61,250
<b>Design</b>	91.9%	\$64,396	8.1%	\$55,156
<b>Audio</b>	91.9%	\$73,764	8.1%	N/A
<b>Production</b>	83.4%	\$79,970	16.6%	\$72,398
<b>QA</b>	95.0%	\$39,309	5.0%	\$34,375

## FOR STARTERS

**AVERAGE SALARIES ACROSS ALL LEVELS OF EXPERIENCE BY HIGHEST EDUCATION ATTAINED**

	PROGRAMMING	ART	DESIGN	AUDIO	PRODUCTION	QA
<b>ASSOC. DEGREE</b>	\$79,375	\$61,071	\$53,500	N/A	\$68,587	\$37,750
<b>SOME COLLEGE</b>	\$87,147	\$69,254	\$63,289	\$70,909	\$84,881	\$31,582
<b>BACHELOR'S DEGREE</b>	\$79,241	\$66,715	\$64,046	\$71,944	\$75,833	\$43,500
<b>SOME GRADUATE</b>	\$86,286	\$82,045	\$71,500	N/A	\$81,071	N/A
<b>MASTER'S DEGREE</b>	\$90,545	\$70,403	\$64,929	N/A	\$87,803	N/A

**AVERAGE SALARIES BY DISCIPLINE FOR DEVELOPERS WITH 3 OR FEWER YEARS' EXPERIENCE**

<b>Programming</b>	\$60,296
<b>Art</b>	\$43,500
<b>Design</b>	\$46,184
<b>Production</b>	\$52,763
<b>Audio</b>	\$53,971
<b>Quality Assurance</b>	\$28,556

**PERCENT OF SURVEY RESPONDENTS WITH THREE OR FEWER YEARS' EXPERIENCE BY DISCIPLINE**

<b>Programmers</b>	29%
<b>Artists</b>	25%
<b>Designers</b>	29%
<b>Producers</b>	13%
<b>Audio Personnel</b>	17%
<b>Quality Assurance</b>	44%

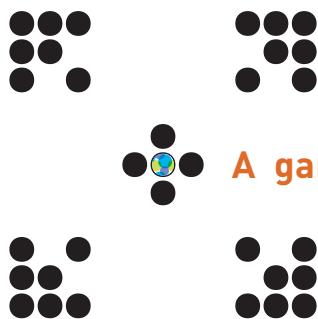
» brandon van slyke

# HOW A GAME GETS MADE

WORKING IN THE GAME INDUSTRY IS A VERY POPULAR CAREER CHOICE—AND rightfully so. What other occupation allows you the opportunity to be on the frontline of technology, creating entertainment experiences that people will interact with and enjoy the world over? Not too many, in fact you can probably count them on one hand.

It's not all fun and games, though. Developing console games is hard work and requires serious persistence and a ton of dedication. The process takes blood, sweat, and a metric ton of Mountain Dew, and is reliant on the skills and expertise of countless professionals whose own passion for games drive them to create fun and rewarding experiences.

How exactly does a game go from the cavernous depths of a game designer's imagination to the shiny cellophane covered package you find lining store shelves? If you try and imagine it like an assembly line with each station in the chain representing where a core piece of the development



## A game's journey from concept to store shelves



process gets integrated, game development is similar in that certain steps must be accomplished before work can begin on the next. From high concept to retail shelf, let's find out how a video game gets made.

### PREPRODUCTION

Preproduction encompasses the planning stages of development and is the time when ideas are expanded upon, designs get fleshed out, prototypes are built, and decisions are made that will affect the project throughout development. It should come as no surprise that preproduction is far and away the most creative phase of a game's lifecycle. A lot of the work done in this initial period is thrown away, but it's necessary waste needed to determine what direction the game will ultimately take. It's during this time that the team really hunkers down and decides on the core elements of the game.

We often read about games that were rushed into production. It's a common occurrence in this industry, caused by a variety of factors, including

---

**BRANDON VAN SLYKE** is a professional video game designer with experience designing commercial console games, mobile games, and casual downloadable web games. He is currently employed as a game designer at Vicarious Visions (an Activision studio) in New York. Email him at [bvanslyke@gdmag.com](mailto:bvanslyke@gdmag.com).

# HOW A GAME GETS MADE

the need to release a game the same day as the movie it's based on. Another reason a game might be rushed into production is that the publisher has a limited amount of time before its rights to a certain license expire, or to hit a big holiday sales period. Despite these kinds of circumstances, it's generally understood that all games have some kind of preproduction phase, in one form or another.

During preproduction, the project's team size is very small and is primarily made up of each discipline's individual leads. The number of people working on a video game project changes during the course of the game's timeline, based on need. The kinds of things that happen during preproduction don't require a full team of programmers, artists, audio engineers, tools creators, and so forth.

Successful game companies try to always have a number of projects going at different stages of development, which allows them to reshuffle their employees appropriately. As one project ends, the production phase of the next one is just starting.

For the first few months of a new project a game designer's time is spent creating flow charts to demonstrate in-game pacing, working on exhaustively detailed rule sets, fleshing out the game's narrative, and comprehensively documenting all the objects, characters, levels, enemies, and NPCs that need to be created for the game. With the majority of their time being spent working on documentation, designers must also find time to meet up with other members of the team who are busy with their own tasks.

Artists, for instance, spend most of their preproduction time fleshing out character designs, creating concept art for levels, and storyboarding scripted in-game sequences. Because they're ultimately in charge of the look and feel of the game, it's important that they also interact regularly with the game designers and writers to make sure that important design considerations are being factored into their work.

Programmers will often use their time during preproduction to create tools needed for development, while also documenting all the technical specifications for the project, which will cover both the tools they are creating as well as any issues and problems they foresee with implementation.

## WHERE DOES THE IDEA COME FROM?

Let's take a step back and look at where an idea for a game comes from. The initial spark can originate from just about anywhere and can be influenced by pretty much anything: classic novels, a leisurely stroll through the park, or a favorite movie on the 37th viewing. More important than the actual idea, though, is how well it lends itself to interactivity. Finding the right context in which to frame your idea is a key part of moving forward with its implementation.

It's a common expression in the game industry that good ideas for video games are a dime a dozen, and in the case of today's larger console titles, popular franchises, recognizable intellectual property (IP) and big name licenses will more

**“Preproduction: Finding the right context in which to frame your idea is a key part of moving forward with its implementation.”**

often than not win out due to previously established market awareness. This is why you tend to see far more sequels to successful titles and a lot of games based on hit movies and television shows; they simply provide lower risk to the publisher.

Where does this leave new ideas? Historically, original IP has come from one of two places: an independent studio that is able to self-fund the project or a publisher's most successful teams who have spent years proving themselves to their risk-adverse owners.

With that out of the way, the question then becomes, who decides what game gets made? In the case of an independent studio, that honor goes to the owner. If it's a mod group, the entire team may be a part of the decision. It's really based on the dynamic of the group.

On the other hand, when the studio is under the umbrella of a large publisher, the studio has the ability to put

dibs on a popular IP or

pitch its own unique idea. At this point, factoring in the state of the market and the track record of the studio, the publisher decides on whether an idea gets a green light.

*Conceptualization.* The core purpose of preproduction is to allow the development team to plan out every detail of the project and outline the production schedule based on the time estimates provided by each department. When the title is a sequel to an existing franchise, the length of preproduction is shortened. The emphasis then becomes figuring out what features will be added and evaluating existing data to see what is already available to work with. It's at this point that new artistic directions are experimented with while determining which path to take.

*Planning it all out.* When it comes to planning a massive project like a AAA console game, producers can be a very valuable resource to have. It's their job during the preproduction process to outline a production schedule based on time estimates provided by members of the development team. If the game is a sequel to an already existing franchise, the emphasis is on figuring out what additional features are going to be added or changed and whether existing data will be reused or if all assets will be created from scratch.

To properly plan, a producer starts by creating an in-depth schedule using a program like Microsoft Excel or Microsoft Project. It's here that they break down each individual task and identify the key points in the project, called milestones. Milestones are fixed dates in which the developer must deliver agreed-upon work to the approval of the publisher. There are actually a number of ways producers can go about scheduling a project. One of the more popular methods, and one you will find is being adopted by a lot of professional studios, is called agile development (see the sidebar "What is Agile Development").

*Prototyping.* Once initial planning has been completed, the team begins working on a prototype. A prototype is a rough mock-up of the game that can be played to see if the design mechanics work together when they're actually in motion. Most prototypes use placeholder art, require minimum assets, and are put together in very short periods of time. More often than not, the work done during this phase is tossed out when



## HOW A GAME GETS MADE

actual production begins because it's important not to leave any artifacts in the code that may introduce problems later on down the line.

Because prototyping takes place in the preproduction process and is just a test bed, it is often done in tandem with the initial design phase to quickly try out new mechanics and ideas to see if they'll work in relation to the game as a whole. Once the prototype has been completed, it's often demoed to the publisher to gain their confidence and to get the actual production of the game green lit.

Once the prototype has been completed and the project has been green lit, the team is able to move on to the next stage of the development process: production. This is when things really begin to heat up, when teams grow, and communication between everyone involved becomes increasingly important.

### WHAT IS AGILE DEVELOPMENT?



Agile development is one of the most popular development methodologies used in the game industry, if not the most popular. Agile development is characterized by modularity and a frequent review of the state of the project, hence giving the team "agility" to easily and quickly change direction if something isn't working. At the same time, it provides product owners with the information they need and the ability to see progress as it's being made.

How it works is that the individual tasks needed to complete any given feature are broken down and assigned to different members of the team who then estimate the amount of time it will take to complete them. Throughout the creation of the game, the team works in "sprints," which usually last anywhere from two to four weeks. During sprints, producers are able to gauge the amount of time a certain feature is taking to implement and can cut or reduce scope as needed.

Agile in and of itself is a fairly complex system and entire books and articles have been written about the benefits it provides. It's far too large a topic to discuss in this article. If you're interested I highly recommend doing a bit of research on it as you may be introduced to it when you get your first job in the industry.

within a large team. This is where production methodologies come into play. A production methodology is simply the process a development team uses to keep track of, and divide up all the work that needs to be done.

### PRODUCTION

When an idea for a game has finally gotten the green light, it's time to begin developing it into a full release title. But where do you begin?

Just as construction companies don't start erecting a building by randomly pounding nails into a board, game developers don't begin working on a game without a solid foundation or plan. All those prior months of preproduction now show their worth as the team begins to build the game to spec based on the design document.

That being said, when making video games, no matter how preplanned and thought-out the preproduction phase seemed to be, things can and will change. Making a game is a largely iterative process with new ideas often implemented on the fly. It's obviously not a free for all, but most studios encourage some form of structured experimentation as long as it can be demonstrated to improve the game.

*Production methodologies.* One of the most important aspects of production is keeping track of everything. Being able to maintain visibility among not only the development team, but also the production staff helps keep everyone on the same page and makes it easier to quickly rectify any issues as they crop up. However, issues and dependencies can be difficult to track

*Building the game.* Up to this point we've discussed where the idea comes from and how planning and prototyping the concept are necessary steps to getting the game off the ground. Now it's time to take a look at how the game is built.

To keep production rolling smoothly and to minimize potential blockers, dependencies must be identified early on. Recognizing which tasks need to be completed before work can commence on another allows developers to properly stagger the workload and production pipeline accordingly. Identifying dependencies also gives the design, art, and engineering teams an opportunity to work independently of one another, effectively maximizing everyone's efficiency.

To kick things off artists begin by creating and texturing character models that the player will interact with as well as the unique props that populate the game world. These objects include items such as cars, buildings, and oh-so-popular wooden crates. When a character model is complete, it's then passed off to animators, who rig the skeletons and generate all the different animations players see in-game. Because art and asset creation is the most time-consuming part of the process, artists usually must get a head start on all of this while the design and engineering teams lay out the core functionality and data that will drive everything behind the scenes.

At this point, level designers are busy creating rough passes of each level in the game. This includes tasks like determining spawn points for enemies, deciding where scripted events will occur, plotting out AI pathing nodes, and demonstrating cool ways that geometry can be incorporated into gameplay. Once they have all of that fleshed out, the level designers begin populating their grey-boxed levels with the custom props that have been recently created by the art team.

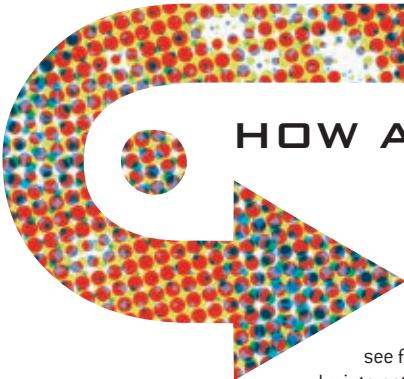
On the data and engineering side of things, the team is busy putting everything together. It's here that system designers start hooking up all the animation and FX data so that characters and entities show up in-game. If need be, the designers will sit down with the animators to tweak animation timing to ensure that it matches the specifications outlined during preproduction. It's also a systems designer's responsibility to tune gameplay and balance the values that define everything from the number of hit points the player's avatar starts off with to the amount of damage a point blank area-of-effect radial attack delivers.

Over in the land of audio, sound designers are kept busy sampling sound effects for different event triggers in-game and acquiring and directing any voiceover talent that is needed for character dialogue or narration. As might be expected, they're also in charge of composing the game's musical tracks and score. Audio is often an overlooked discipline that should not be taken for granted. Their contribution to the game is very important and their work will not only bring the game to life and help set the mood, but can also be used to aid the design.

*Don't break the build!* If you have a team of 70 people all working on the same data, how is it that nothing gets deleted, overwritten, and ultimately destroyed? I'll admit it can be a fairly chaotic process. However, there is a solution.

Development teams utilize version-control software to help ensure that any changes made to the current build, like a variable in code or a value in data, doesn't break what's already

# HOW A GAME GETS MADE



there. Version-control software is great because it allows developers to overwrite, redo, and add changes as they see fit. If there is a conflict, it can be resolved by interactively merging each team member's

version of the file. The whole system provides developers with constant updates and an impressive amount of visibility, effectively guaranteeing that no work is accidentally lost.

*From part to whole.* At various times throughout the project, the team should get together to review and critique the most recent build. It's at these points that the people working in the trenches can take a break and assess the game as a whole, allowing them to see what their teammates have been up to the past few weeks.

Not only does everyone get to have their work constructively critiqued, but they can also point out and discuss their contributions and progress, which might not have any visible effects on the game. For example, if someone made a new effects-rendering system or wrote new documentation outlining a narrative for a cut scene, their contribution might not be readily apparent—but it is important, and these meetings allow everyone on the team the chance to acknowledge those kinds of contributions. More than anything, though, these meetings are meant to maintain open communication among the team and enlighten everyone on the current status of the project.

*When does testing start?* Quality assurance (QA) testers are usually integrated toward the middle of production, although leads are on the project from the beginning. During the beginning of production, they're often busy getting the bug tracking software set up and testing each build to make sure all known issues are recorded and assigned to the proper person on the development team.

At about two-thirds of the way through production, additional staff will be brought on to fully test the game. It's at this point that the team should have the majority of features implemented. Now it's time to refine everything and squash any and all bugs that come your way.

*Home stretch.* Once the game reaches alpha (a near complete phase), things will seem to be shaping up nicely. The staff will report bugs at a record pace, and the developers will be resolving them just as fast. The only item left is to market the game and get it on store shelves!

## POSTPRODUCTION

Congratulations, the game is complete. That must mean all the work is done, right? Well it is—for the development team. It's now time to find out who steps in and what happens after the game is finally made and on its way to store shelves.

*PR and marketing.* While the developers have been toiling away designing boss battles and drawing up finite state machines, the marketing staff on the publisher's end has been busy figuring out how they will position the title once it's complete.

Marketing sometimes has significant influence on how well a game is understood by both critics and fans, and their work is often

the first thing to make an impression on a potential customer. For instance, they coordinate exclusive reveals with industry magazines and popular video game web sites, as well as manage the deployment of television ads and tie-ins with other consumer products.

*Publishing and distribution.* Along with funding the development of the game, the publisher also handles regional localization, creation of the game's manual, and orchestrates manufacturing of the final packaging. When the final build of the game is completed (that version is called the gold master disc) and sent off to be printed, the publisher has usually lined up a distributor to handle getting the game onto store shelves.

Once the game has been printed and orders have been made, the game is ready to be shipped to your local retailer, where it's hopefully enjoyed by everyone who's decided to check it out.

## WAIT, THERE'S MORE?

Even when a new game is finally released, it sometimes isn't finished. With the rising number of high-speed internet enabled consoles and the ever-growing adoption of digital distribution, developers are now regularly being tasked with creating additional downloadable content to help extend the life of their titles. Downloadable content can include anything from additional characters and costumes, to new level maps, and even interviews with the developers. A lot of the people who are hired during the course of the project are likely to be retained to create these additional features.

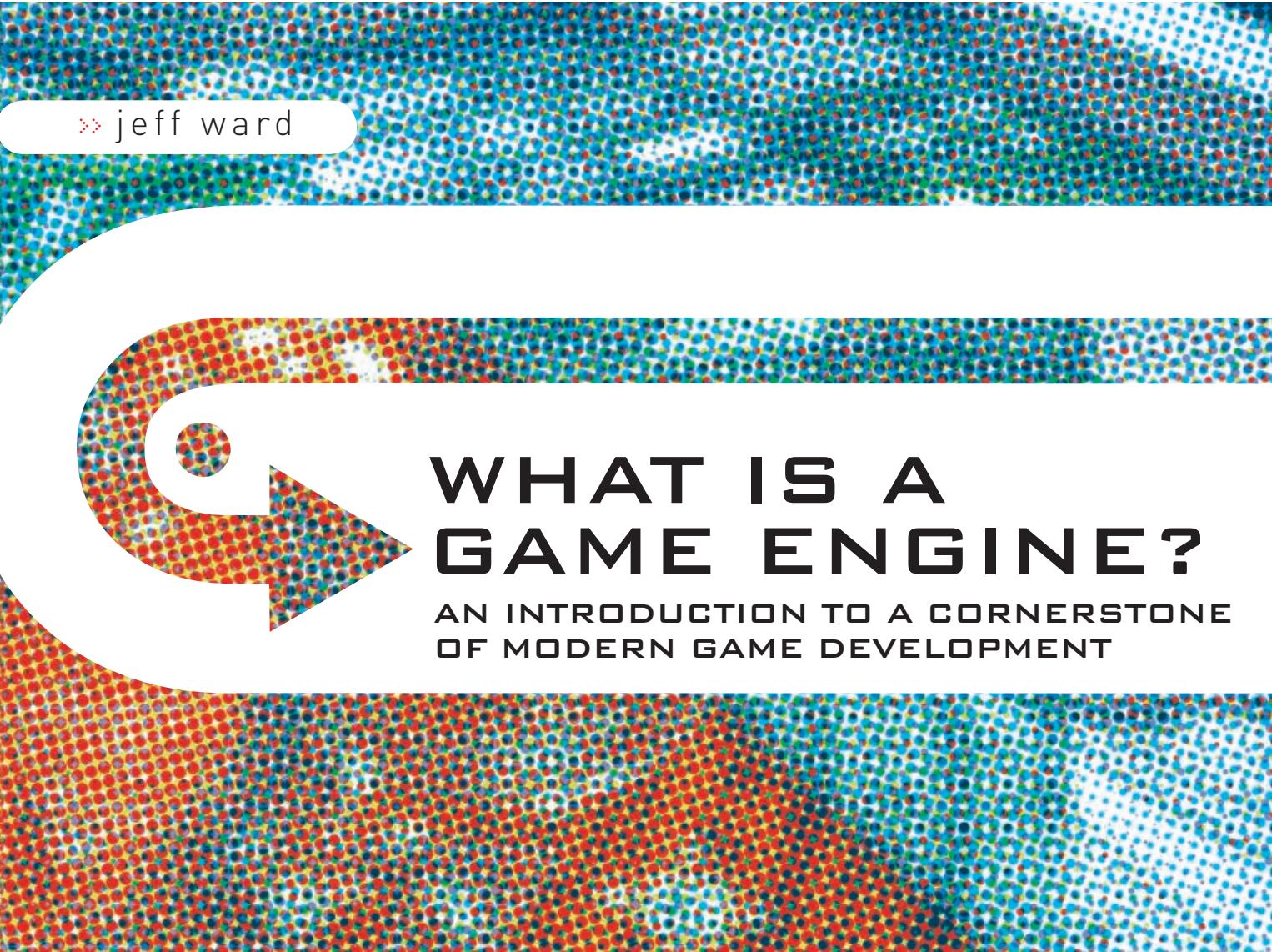
If the work gets outsourced, it's important that the studio developing the downloadable content has the ability to

communicate with members of the original development team. This is especially true as issues arise and they learn to work within the constraints of the data. A wiki is often created as a way of chronicling important need-to-know information

and is usually made accessible in situations like these.

Earlier I mentioned the fact that team sizes fluctuate—what happens if the company doesn't need you any more when it's working on the downloadable content? What happens when your role on the project is finally complete? Throughout the project, and especially near the end, tasks and roles come to an end and it's time to start work on the next one. It's at this time that managers and team leads reshuffle staff and place them either on the studio's next game, move them laterally to another project they have in development, or in more unfortunate cases lay them off.

And there you have it! That's more or less how a modern day console game gets made. Now that you understand a little more about the process of bringing a game from concept to completion, it's important to reiterate that teamwork, communication, and proper planning are essential to creating a game in today's competitive environment. Now what are you waiting for? Get out there and make some games! ☺



jeff ward



# WHAT IS A GAME ENGINE?

AN INTRODUCTION TO A CORNERSTONE  
OF MODERN GAME DEVELOPMENT

## IF YOU FOLLOW THE GAME INDUSTRY YOU'LL HEAR THE TERM

"game engine" thrown around a lot. And I bet sometimes, late at night when everyone else has gone to sleep, you sit and wonder, "What is this nebulous thing I keep hearing about?"

You would expect that the answer would be as simple as being shown a car's engine: "Yeup, thar she is." After all, the game engine, much like a car's engine, is what makes the game go. Unfortunately, sometimes there's a fuzzy line between where a game's engine ends and where the content of a game begins, as if there were a fuzzy line between whether a car's air conditioner is part of its engine.

Generally though, the concept of a game engine is fairly simple: it exists to abstract the [sometimes platform-dependent] details of doing common game-related tasks, like rendering, physics, and input, so that developers [artists, designers, scripters and, yes, even other programmers] can focus on the details that make their games unique.

Engines offer reusable components that can be manipulated to bring a game to life. Loading, displaying, and animating models, collision detection between objects, physics, input, graphical user interfaces, and even portions of a game's artificial intelligence can all be components that make up the engine. In contrast, the content of the game, specific models and textures, the meaning behind object collisions and input, and the way objects interact with the world, are the components that make the actual game. To use the car analogy again, think of how the body, CD player, in-dash navigation system, and leather seats make the actual car. That's the content.

## APIS AND SDKS

Two other terms you hear in the game industry that are closely related to game engines are "API" [application programming interface] and "SDK" [software development kit]. APIs are the software interfaces that operating systems, libraries, and services provide so that you can take advantage of their particular features. An SDK is a collection of libraries, APIs, and tools that are made available for programming those same operating systems and services. Most game engines provide APIs in their SDKs. The Unreal Engine, for example, provides an interface for programmers to create their games, both through a scripting language called UnrealScript, and through libraries, which are provided to anyone who licenses the engine, and which come in the same package as their other tools, like the editor UnrealEd.

But where do game engines come from?

## BIRTH OF A GAME ENGINE

For a long time, many game companies made their own game engines and kept that technology in house, iterating on it as computers improved and more advanced versions were needed. Engines like SCUMM by LucasArts and SCI by Sierra, for example, powered most of the adventure games that those companies released in the late 1980s and into the mid 1990s. More recently, engines like id Tech [the engine that powers the QUAKE series of games] and the Unreal Engine started as in-house technologies, though they have recently evolved into middleware technologies as well.

---

**JEFF WARD** is a co-founder of and lead programmer at Orbus Frameworks, a provider of metrics and data-gathering tools for game developers. Email him at [jward@gdmag.com](mailto:jward@gdmag.com).



## WHAT IS A GAME ENGINE?

Over the past several years, the cost of making an in-house engine has grown significantly, and more and more companies have begun to specialize in making either full game engines or game engine components to sell to other companies, rather than make games. We call these kinds of companies middleware providers. The middleware providers can offer these products at very reasonable prices, and, for most game development studios, this creates a very clear "build versus buy" decision. Why pay six programmers for a year to build an engine when you can buy 90 percent of the features you want from a proven technology for less money—and have it immediately? As a result, almost all components of a game engine are purchasable at a variety of prices, or downloadable in the form of open source projects.

### TYPES OF GAME ENGINES

Game engines come in many different flavors and at many levels of programming expertise. To get a feel for how different they

can be, I'll explain three kinds of engines: the roll-your-own version, the mostly-ready version, and the point-and-click engine.

#### *Roll-your-own game engines.*

Despite the cost, many mainstream companies (as well as indie game makers) will still roll their own engines. This means they use publicly available application interfaces, such as APIs like XNA, DirectX, OpenGL, the Windows and Linux APIs and SDL, to create their

own engines. In addition, they may use other libraries, both commercial and open source, to help them along the way. These libraries might include physics libraries like Havok and ODE, scene graph libraries like OpenSceneGraph, and GUI libraries like AntTweakBar.

I include XNA and SDL here because, although they make creating an engine much easier by abstracting away some

of the more nasty platform implementation issues, they still require a lot of programming to get a game off the ground. They're not really engines so much as good starting points for creating your own engines.

Generally, these home-rolled systems give programmers the greatest amount of flexibility, letting them pick and choose the components they want and integrating them exactly how they want. But they also take

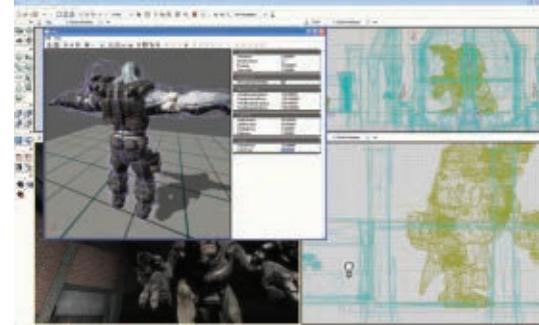
the longest amount of time to build. Additionally, programmers frequently will have to build the tool chain from scratch, since they can rarely rely on all these libraries to work together straight out of the box. This makes rolling your own engine less attractive to most game developers, even the professional ones.



ARID OCEAN utilizes the OGRE 3D engine.



A game environment built with Torque Game Engine.



Editing physics in UnrealEngine3.

*Mostly-ready game engines.* I consider most game engines to be "mostly ready." These engines are ready for prime time right out of the box, with rendering, input, GUI, physics—you name it. Many of them even have mature tool chains so you don't have to roll your own. Engines in this category include OGRE and Genesis3D (which are both open source), low priced engines like Torque, and even really high priced ones such as Unreal, id Tech, and Gamebryo.

To varying degrees, all these engines still require a bit of programming to get them up and running a complete game. They might call for some scripting or sometimes even low-level coding to get a real game working. Mostly-ready game engines are a bit more limiting than roll-your-own engines and are frequently optimized for the general case. That said, many of these engines are the product of dozens of people's work over hundreds of long hours, and will provide better performance with less effort than most roll-your-own engines, even if they don't do exactly what you want.

*Point-and-click engines.* Point-and-click engines are becoming more and more common these days. They include a full tool chain that allows you to point and click your way to creating a game. These engines, which include GameMaker, Torque Game Builder, and Unity3D, are built to be as friendly as possible, and are made to require as little coding as possible. That's not to say knowing a little coding doesn't help, but it isn't really a necessity the way it is for the mostly-ready and roll-your-own engines.

The problem with many point-and-click engines is that they can be extremely limiting. Many do one or two types or genres of game well, or one or two types of graphics modes. This is not to say they're useless. Even faced with the restrictions of these tools, it's possible to make highly creative games or even find creative ways around those restrictions. The best thing about these engines is that they allow you to work quickly, and play your games quickly, without too much work. If you're just starting out in game design, you could do worse than these tools. ☺

### RESOURCES

There are many resources for finding game engines. Wikipedia is generally the best resource for engine information, and it lists all the known open source engines. These pages are frequently updated and give feature comparisons:

[http://en.wikipedia.org/wiki/Game\\_engine](http://en.wikipedia.org/wiki/Game_engine)  
[http://en.wikipedia.org/wiki/List\\_of\\_game\\_engines](http://en.wikipedia.org/wiki/List_of_game_engines)

» jake simpson

# GAME JOB INTERVIEW QUESTIONS

And How to Answer Them

## » YOU'RE AVIDLY APPLYING FOR YOUR FIRST JOB IN THE VIDEO GAME INDUSTRY.

A couple of companies saw your enormous potential in your resume and cover letter, and they want to bring you in for an interview. How should you prepare?

In the game industry, there are several interview questions that tend to come up a lot. Here's a smattering of those and some tips on how you might handle them.

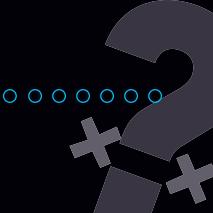
### 1 WHY DO YOU WANT TO WORK HERE?

(This question implicitly includes, "Why do you want to leave where you are?" if you're currently employed.)

This question is an open opportunity to show you've done some research on the company where you're interviewing. All companies and interviewers are flattered when the interviewee knows who they are, knows what games they make, and wants to be a part of their experience. Do your homework and put on a good show!

Don't say things like, "I need a job," or "I need to move to Sacramento." Instead, pick a few things that are germane to the company in question. The more specific your reasons are tied to the company, the better. "I want to work on FPS shooters" isn't as good an answer as "I want to work on *Game Franchise X* because I played the first two games and still see potential for future growth of the product." It's sycophantic, yes, but interviewers are as

JAKE SIMPSON works for Linden Lab creating a safe haven for the weird and the wonderful of all types within SECOND LIFE. Previous to this, he worked on THE SIMS 2, SOLDIER OF FORTUNE I and II, STAR TREK: ELITE FORCE and some Midway arcade games. He did not invent the blue LED nor is he part of the US sky diving team, no matter what he might tell you in person. Jake likes beer so if you meet him, hand him one and back away slowly or he'll talk you to death. Email him at [jsimpson@gdmag.com](mailto:jsimpson@gdmag.com).



# GAME JOB INTERVIEW QUESTIONS

prone to flattery as anyone else—although don't give that as your only reason.

When explaining why you want to leave your current job, the trick is to not be negative. Pick a couple of points that are inarguable, for example, "There was no career development" or "They weren't working on the kinds of games I'm interested in," rather than "Their management is clueless and they are going to die soon." The game industry is a small community—you could very well be talking smack about your interviewer's close buddy.

If you were let go or fired, it's better to say something like, "We decided to part ways," or "It was my time to leave," rather than go into too much detail, unless directly pressed. In that case, the interviewer probably already knows what went down and is just looking to see what you'll say. Answer the question quickly and without negativity, and move on. You want to leave a positive impression.

## 2 WHAT GAMES ARE YOU PLAYING?

If you plan to work for a video game company, you'd better be playing games—and you'd better be able to demonstrate that.

It's good form to mention some games that are in the same genre as the games made at that company. It's even better if you mention playing some of the games that were actually made there. Again though, don't go over the top.

At the very least, play the demo of anything they've produced. You need to be knowledgeable about the genre, what you enjoy about it, and how the development of these games is affected by the genre (as much as you can be). So research the company before the interview.

How you answer this question can be a deal breaker or a deal maker for hiring managers. They want to hire people who are demonstrably passionate about the games their company makes. Saying, "I have a level 70 mage in WORLD OF WARCRAFT and a level 40 druid in EVERQUEST," to Blizzard makes the point that you are immersed in its product genre.

Demonstrating some knowledge about older games also shows you're grounded in game history, which is never a bad thing. The wider your knowledge base, the more you can forestall going down blind alleys in terms of implementation and design, which benefits everyone, and that's exactly what a company is looking for in its employees.

## 3 HOW WOULD YOU MAKE THE GAMES YOU'RE PLAYING BETTER?

You'd be surprised how often this question comes up, even if you aren't interviewing for a design position. Everyone wants a developer who has design sensibilities because it inevitably means she or he will be more involved and engaged in whatever is going on.

Knowing ahead of time how you might answer this question means you'll come off sounding like you've actually thought about a game in development terms. Game studios are looking for people who think as they play—about what they're playing, how it's done, what could have been improved, and most importantly, what they can rip off.

One downside to adopting this mentality is that it becomes harder to enjoy a game for what it is, but that's an occupational hazard in all jobs.

Believe it or not, you can answer this question in an entirely positive way. However, if you decide instead to criticize a design or implementation decision in a game, be sure you have a solution to the problem too. It's not enough to moan about the final strider battle in HALF-LIFE 2: EPISODE 2; you have to have an idea of how it could have been made more enjoyable, perhaps through easier car control, or not destroying all the supply stations so quickly. If you decide to bash a game that the company where you're interviewing developed (and that takes courage; some companies will applaud you while others will diss you for not drinking the Kool-Aid), then ensure that what you're criticizing isn't something subjective but something that everyone has had a pop at. Be ready to back up the criticism with proof that it's an agreed-upon flaw, not just you being nit-picky.

## 4 WHAT'S THE BEST GAME OF ALL TIME AND WHY?

The most important thing here is to answer relatively quickly, and back it up. One of the fallouts of this question is age. Answering "ROBOTRON!" to a 20-something interviewer might lead to a feeling of disconnect. But sometimes that can be

good. It means you have to really explain why it's the best game of all time. Can you verbally and accurately describe a game to another person who has never played it? You'll rack up some communication points if you can.

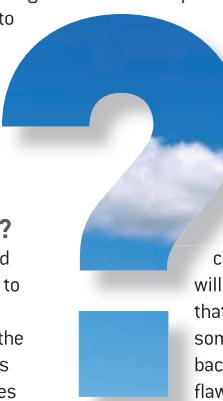
What you shouldn't say is whatever the latest hot game is, or blatantly pick one that the company

made (unless it's true and your enthusiasm is bubbling over). Be honest. Don't be too eccentric and niche, and be ready to defend your decision.

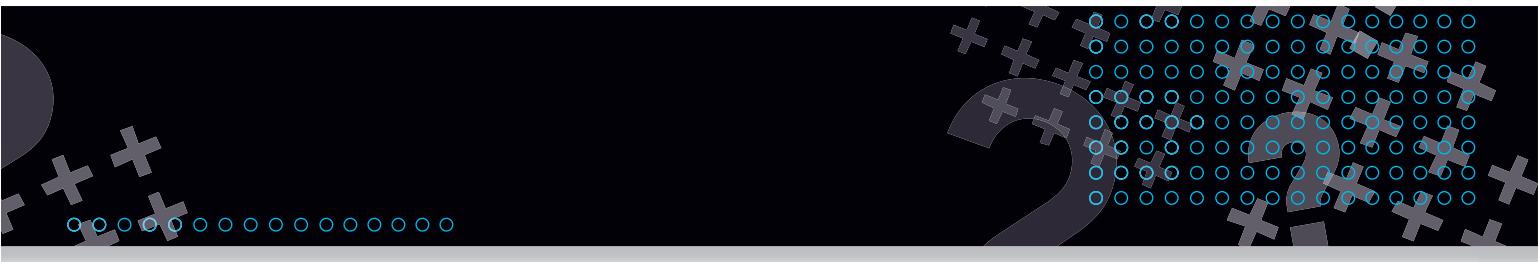
## 5 WHAT WILL YOU BRING TO THE TEAM? WHY DO WE NEED YOU?

This is a general question that applies to all interviews. There are two ways to answer: the big answer and the little answer.

The big answer requires you to have some knowledge of how the company operates. Who does what? Your goal is to slot your experience, passion and skills (and if you are a student, your passion, skills, and desired career direction) into any holes the company may have—and it should have some. Otherwise, why are they hiring?



**The best way to prepare for a job interview is to do some practice interviewing. Students and recent graduates can usually do practice interviews at their school's career center. If you've received a lot of calls for interviews, schedule meetings with the companies you're least interested in first, and consider those encounters practice rounds.**



The little answer is to name some of your previous experiences and best qualities and hope that's enough.

Care needs to be taken that a) you don't sound arrogant in assuming the company will die without you and b) you don't say negative things about the company. Statements like, "Well, you obviously can't do good Q/A. You need a good Q/A manager," are likely to go down like a lead balloon. Frame your answer to suggest that you would bring extra expertise, and therefore improvement, to something that's already in place.

## 6 WHAT'S YOUR BIGGEST WEAKNESS? OR, IF I HIRED YOU, WHAT WOULD I REGRET ABOUT IT IN SIX MONTHS?

This is a common question in all job interviews. There are generally two kinds of responses: the brutally honest and damning one ("I get upset with people who don't carry their load"), and the sycophantic one ("I'm a perfectionist").

What most employers are looking for is an honest answer that is followed up with an example of something you have done to work on your weakness. For example, you can say, "My workspace tends to become extremely disorganized," as long as you follow it up with, "but recently, I've put in a lot of effort to go paperless, and I'm extremely systematic in the way I manage my email inbox."

The other secret to this question is not so much in the answer but how long you take to respond. If you answer too quickly, you might be suggesting that you already know all your worst points because they are blatantly obvious and you've been told so many times. If you take too long, it will seem as if you're searching for an answer that sounds good, doesn't make you look bad, and is something the interviewer would be happy to hear. Again, it gives the perception that you are being ingratiating rather than honest.

By the way, the best answer I've heard is, "I don't know. What do you think I'd regret in six months if I worked here?"

## 7 HOW DO YOU FEEL ABOUT CRUNCHING?

At smaller studios, this is the 64 million dollar question. My advice is to be 100 percent honest. If you won't crunch, say so now. It may well put you out of the running for a job, but ultimately that's a good thing. No, really, it is! If the company works a lot of overtime and you don't want to do it, then taking the job is going to be punishing for everyone.

Having said that, the last thing any interviewer wants to hear is, "I won't do it" because that predicates a perceived lack of involvement and passion (not that passion should equal overtime, but the perception of refusing to do something before you're even in the circumstances could be the difference between getting a job offer and having the company pass you up).

Phrase your answer in such a way that you don't sound confrontational with the interviewer. She doesn't want to get into an argument; she just wants to know where you stand. Understand that this question is meant to gauge, roughly, how you might fit into the company culture.

## 8 OKAY, WE'RE GOING TO WORK THROUGH A PROBLEM HERE ...

Often in game job interviews, you will be presented with a problem to solve, or even a full-blown test, depending on the position. It might be grease board work, it might be a conversation, it might be a level design test, it might even be a code test at a PC.

The premise is that the interviewer wants to see how you work. Often, once you've answered the question, the interviewer will change the parameters to see what you'll do.

But what do you do if you have no clue what's being asked, or if it's outside your area of expertise? That's a panic moment if there ever was one. Take a deep breath and realize that this is a moment where you need to say, "I'm not sure I understand the question," or "That's not something I've done before." But immediately after that, start asking questions about the problem and take a stab at solving it.

That's one of the biggest things you can do at this point—admit ignorance then have a go anyway. Showing a willingness to try something outside your field of knowledge is huge to interviewers. It shows you want to learn and be more than what you are now. Sometimes, the fact that you tried is more important than the actual result, and sometimes, you'll have an interviewer who will give you hints toward a solution just because you showed that willingness to try. The more junior you are the more likely this is to happen.

Occasionally, interviewers will deliberately put you out of your comfort zone just to see how you'll react, so be aware!



good  
lets  
you  
in



Q: future generation game middleware from the creators of Direct3D:

- High-end runtime and tool features
- Extensible plug-in framework: cleanly customize anything
- For all platforms and any genre
- Share technology across games and license it to other Q developers

Try Q free today. For more information, contact Jamie Fowlston on +44 (0) 20 7431 9995, email info@qubesoft.com, or visit www.qubesoft.com.

# GAME JOB INTERVIEW QUESTIONS



## 9 WHERE DO YOU WANT TO BE IN FIVE YEARS?

Personally, I love this question because it reveals if a prospective candidate has a plan at all or is just drifting from job to job as so many are wont to do. There's nothing wrong per se with people who drift along the currents, it's just that those with a plan (or at least a desire to move in a particular direction) are generally much more interesting people. Plus, they are almost always inherently more predictable, which is always a benefit for employers.

Having a desire to move forward helps everyone. It helps you measure your progress, and it gives the company a plan to help you get there.

Of course, it does depend on you knowing what you want. Most people tend to know what they don't want, but not necessarily what they do want, which is a problem—particularly if you express that in an interview. Interviewers would rather have a list of things you want to attain rather than things you don't.

One optimal answer is, "Still working for you making games," but it smacks of sucking up, so I'd recommend saying something a little more generic: "Still looking for a challenge and putting in that extra effort to make great games."

The best response I've ever heard to that question was, "I want your job!" and the individual who said it to me indeed has my old job! But be wary of sounding confrontational.

## 10 WHAT GAME WOULD YOU MAKE IF MONEY WERE NO OBJECT?

Everyone has a pet project they would want to make if they had the chance—it's just inherent in the game developer psyche. This is your chance to expound on it, and the more realized your idea is, the more it will be seen as proof that you know what you're doing.

Taking an existing idea and adding, "but I'd make it cooler!" isn't the answer (the number of times I've heard Q/A staff wanting to become developers tell me they want to remake COUNTER STRIKE "but better" is staggering); it just shows you have enthusiasm, but no original ideas.

Bonus points if you can take an existing IP license and make a compelling argument for a game out of it. People who can

actually do that are at a premium in our industry since most tie-ins, well, suck.

## 11 WHAT DO YOU DO ON YOUR OWN TIME TO EXTEND YOUR SKILLS?

As a programmer, do you work on home projects? As a designer, do you doodle design ideas or make puzzles? As an artist, do you do portrait work?

Having hired many people in the past, one of the things I can speak to with authority is that those people who spend their off time working on discipline-related projects are the ones who are always up on current trends, have new ideas, are most willing to try something new, and will be the ones taking stuff home to tinker with on their own time. Now that shouldn't be expected of everyone, but the sad reality is that there is competition for jobs out there, and those who are prepared to put in the extra work are the ones that are going to be in hot demand.

Demonstrating that you learned C# over a weekend because you thought it was cool for prototyping is exactly the kind of thing a programming manager wants to hear. Suddenly your toolset expanded, and not only did it show willingness to do something without being told, it makes you more valuable.

The only care to here is to not mention an outside situation that might detract from or compete with your day job.

## 12 DO YOU HAVE ANY QUESTIONS?

Yes. Yes, you do have questions. Some of your questions will have been answered in the normal give-and-take of conversation, but you should always be asked if you have others (and if not, something's wrong).

Having questions means you're interested. Some questions are best directed to HR, while others should be asked of managers and future co-workers. Ask questions that show an interest in the position and the long-term plans of the company. For some ideas, see "Questions You Should Ask in an Interview."

## IT'S JUST A CONVERSATION

Always have something to say. Most questions asked in job interviews aren't knowledge-test questions. They're more like an invitation to converse on a given topic. If you can't answer them directly but can at least talk about what the question means, that's a good start.

Be as honest as you can be. A company generally gets only a day with you, maybe two, to gather all it can before making a life-changing decision, both for you and for them.

They need to see who you actually are rather than whom you think they want to see, because if that's what you do once you've started, it will become obvious pretty fast, and then bad things tend to happen. They need to be comfortable that there are no surprises coming with you as much as you need to know that they are solvent and able to actually ship what they say they will.

Be honest. Be open and engage in the conversation with your interviewer. Express interest in what they are talking about. If you don't know or understand what's being asked, ask them! They don't expect you to be mind readers and it shows you can ask when necessary. After all, it's just a conversation. ☺

## Questions You Should Ask In An Interview

- \* What are the core working hours?
- \* How do you assign or schedule tasks?
- Who gets to decide who does what and estimates time?
- \* What's the career path for this job? How do I get to progress? What is the process for promotion?
- \* What training approach do you use?
- How would I learn new skills?
- \* How are personnel reviews handled?
- Who does them and how often?
- \* Are there any specific development processes used here, for example, Scrum?
- \* Who would I report to?
- \* If I'm hired, what is the next game I might work on? How much input would I have on that?
- \* Is there a relocation package?
- \* What bonus structure or incentives are there?

**POSTMORTEM**

# GESUNDHEIT!

a student made game



**IN MY FINAL TERM OF SHERIDAN COLLEGE'S** illustration program in Ontario, Canada, I took a class in which each student had to propose and develop his or her own project. While this assignment obviously had to include some illustration, the form of the final product was up to the individual. Suggested projects included books, posters, and advertisements, but for some reason I had become obsessed with the idea of making a computer game. At first, my goal wasn't very sophisticated—the gameplay would just be an excuse for me to animate little monsters eating each other.

A year or two earlier, I had created a few little point-and-click adventure games using Adventure Game Studio (a free engine for creating 2D KING'S QUEST-type games), and in the summer before my final four-month term, I had been testing some action game concepts with the same engine. When classes started, I was able to show my illustration teacher Harvey Chan, a rough prototype of the game I wanted to make, and he encouraged me to follow through with it on the condition that I give it some decent artwork. "Make it look like your drawings," he said.

With that advice, I jumped fully into production on GESUNDHEIT!, a 2D overhead action-puzzle game with single-screen levels, hand-drawn graphics, snot-eating monsters, and a sneezing pig.

#### WHAT WENT RIGHT

**1 HAVING A ONE-MAN TEAM.** Obviously, making a game with a team of one meant that there would be severe limits to the scope of my project. But seeing what other soloists had done with the AGS engine—Ben "Yahtzee" Croshaw's games are a great example—was very encouraging. Besides, I knew that CAVE STORY was a one-man show, and if Pixel could do it, then so could I!

In fact, there were indeed a lot of nice things about me being the entire team. I didn't have to worry about any conflicts of vision, and nobody felt like their views were being ignored. There was no miscommunication, either—if some animation took longer than

expected, I wouldn't have to explain it to a programmer.

I was lucky in that I already had a bit of experience in some useful areas. I'd lately been making pixel animations for fun. I had done some QUAKE mod animation back in high school, and I'd been playing and recording music with amateur rock bands for years. Also, in the making of my last point-and-click game, I began to learn the AGS scripting language.

I hadn't a clue how to make an action game, and I'd never done any real programming before, but with the help of the AGS forums, I felt I could figure it out. Besides, I didn't really have the option of working with a team. The assignment was an independent project, and my artist classmates weren't too inclined to venture into the technical sludge of making a game.

It was only after I had posted my first release of GESUNDHEIT!—and again later at the Independent Games Festival—that I thought how nice it would have been to have somebody to share the process with. Knowing a programmer would have been convenient, too, but really, it never once occurred to me that my project required another person.

#### 2 FINALIZING THE GAME MECHANIC FIRST.

Because I was working alone, I couldn't afford to spend too much time on things that didn't directly serve the end product. I needed to have a simple and easy-to-make game mechanic that could provide a reasonable amount of gameplay on limited assets, because I could only generate so much artwork. Even though I originally only wanted an excuse to make graphics, I knew the gameplay had to be settled first so that I could focus on assets that would actually be needed.

I had a pretty good idea from the start about the kind of game I wanted to make. I had loved LEMMINGS for its cleverness, cuteness, and goriness, and I liked figuring out some of the overhead puzzles in the 2D ZELDA games and GOD OF THUNDER. I was mostly thinking of those kinds of spatial puzzles when I did my first prototype, but I guess a little bit of METAL GEAR SOLID snuck in subconsciously.

**MATT HAMMILL** is an illustrator, animator, gamemaker, and karaoke superstar. He likes dinosaurs and dogs. Email him at [mhammill@gdmag.com](mailto:mhammill@gdmag.com).

#### GAME DATA



**SCHOOL** Sheridan College, Ontario, Canada

**DEVELOPMENT TIME** 4 months in school plus 3 months part time before release

**TEAM SIZE** 1

**HARDWARE USED** Laptop PC, Wacom tablet, USB audio mixer, instruments and art stuff

**GAME ENGINE** Adventure Game Studio

**SOFTWARE USED** Photoshop, Anim8tor, Cubase, MS Paint

**TOTAL LINES OF CODE** About 6,000

**FINAL PROJECT GRADE** A

**DOWNLOAD** [www.underwaterbase.com](http://www.underwaterbase.com)





Gameplay concepts were sketched out before coding.

My original build featured a sneezing character (at the time it was an old pixel drawing of my girlfriend) that shot boogers to lure snot-eating enemies through maze-like levels. The puzzles were based on line-of-sight, and the challenge was to lead the monsters through the maze into traps while keeping yourself hidden. If there were no boogers to eat, the monsters attacked.

I had this whole concept drawn out in my sketchbook before I began my first line of code, and although it needed to go through several iterations before I started finding the fun in it, the final gameplay is pretty faithful to my original sketches. (It no longer stars my girlfriend, though; the main character became a green pig with prominent snot-launching nostrils.)

Not being a programmer, it's hard for me to toss off gameplay tests, so I was lucky that I liked my original concept for GESUNDHEIT! well enough to see it through production. And tweaking the gameplay, far from being a chore, was actually quite interesting and enjoyable.

**3 USING THE AGS ENGINE.** Chris Jones' AGS engine was an enormous boon to my development process. Without it, there's absolutely no way GESUNDHEIT! would exist. In fact, the only reason I began making games in the first place was because of how quickly I could pull some animation frames into the engine and see my own characters walking

through crudely drawn backgrounds. It's wonderful that, thanks to engines like this, an art student like me with no technical background can put together a game.

Of course, there are limits to what AGS can do, but when I was starting GESUNDHEIT! I had no problem with that. The low res (640x480) 2D graphics didn't bother me because it kept the sprites manageable and the download size small. The single-screen backgrounds worked fine for the line-of-sight gameplay, too, because I didn't want to worry about monsters being able to spot the player from beyond the edge of the screen. Also, AGS has a great pathfinder for point-and-click adventure games (so your hero can find his way around a table, say) and I used this extensively for both the player control and the movement of the pursuing monsters. I barely had to think about pathfinding at all when I was designing the levels.

**4 ROUGHS, ROUGHS, ROUGHS!** One thing that was stressed throughout the illustration program was the importance of rough work. The idea of using quick little thumbnail sketches for problem solving was drilled into my head for over four years, so I approached the game with the same method.

My sketchbooks are full of level design drawings. The slow-paced, strategic nature of my game meant that I could roughly play through my maps, with the help of scribbling monster paths and lines-of-sight overtop of my drawings, without ever needing to turn on my computer.

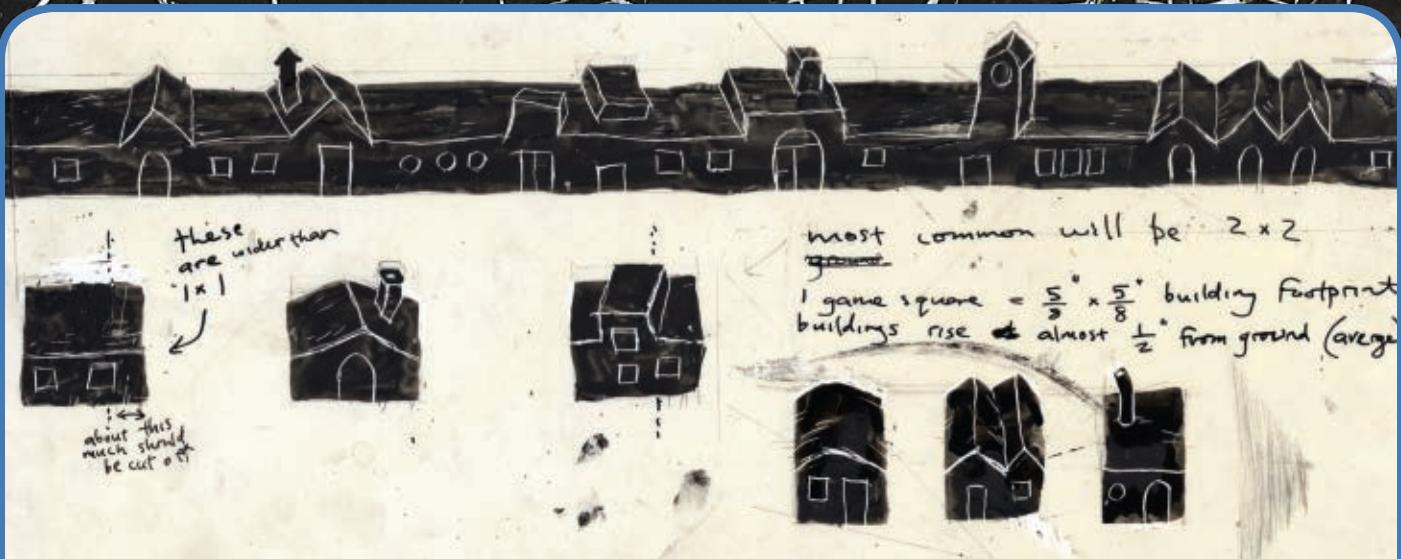
Drawing on blank white paper was much faster and more fun than trying to figure out the levels pixel by pixel on the screen. There was also no pressure to hang on to bad designs because there was hardly any work invested in them in the first place! An additional bonus was that working in my sketchbook meant I could be productive even during the long bus rides to school.

As for the character graphics, using quick and dirty Microsoft Paint sprites as placeholder art early on saved loads of time in the long run. This helped me determine the size of my characters, the required list of animations and their durations, the necessary level assets, and the technical feasibility of my game, all without too much invested in art. Inevitably, there were some changes to be made even after the final assets were created, but I never had to throw out a painstakingly animated loop because it was no longer needed.

One more important step I took before doing my final artwork was to create a mock screenshot in Photoshop.

Here I could see how the sprites and background art would look together, and I could tweak things quickly, without worrying about technical stuff. That fake screenshot became a standard for me to work toward.





Level art was created on scratchboard and then scanned into Photoshop.

I've gone through a few abandoned AGS projects where I've invested weeks into elaborate artwork only to realize that I didn't have a game to hang it on, and I wanted to avoid that this time.

**5 AN ACHIEVABLE AESTHETIC.** As an illustrator, making the graphics and animation was what had driven me to make the game in the first place, and after that first talk with my illustration teacher—"Make it look like your drawings!"—I was even more determined to keep the aesthetic at the forefront of development.

However, I was aware of the technical limits of both myself and the AGS engine, and I didn't want to aim for something I wouldn't be able to achieve. By starting with hand-drawn artwork, I could hopefully appeal to non-gamers, such as my teacher [I was still looking for a good grade!] and also offer something different to traditional gamers and the AGS community, which is mostly dominated by pixel art.

I had been doing some scratchboard drawings on plastic the year before (painting with black ink on a sheet of plastic, then using a knife to scratch white back into the ink) and I liked the unrefined messiness of it. With that as my starting point, and Photoshop for adding color, I eventually created all my characters and level art. The characters' body parts are scratchboard drawings, the shadows are ink thumbprints, and the dirty white background is a piece of unwashed board.

After deciding against MIDI music at the urging of one of my rock band friends, I tried to carry the same messy, naïve aesthetic into the music, as well. It wasn't a difficult aesthetic to

achieve, considering I could barely play any of the instruments I was recording, but having that goal in mind let me relax about my terrible musicianship.

## WHAT WENT WRONG

**1 NOT LEARNING PROGRAMMING FIRST.** I had never done any programming whatsoever before starting with AGS script. I didn't know what an "int" was, and all those curly brackets and semicolons were very confusing.

Instead of spending a couple weeks going through some basic programming tutorials, I decided that I'd learn by doing, and I just jumped into scripting my game. I had the AGS help file, and as long as I could get stuff working, then what was the problem?

Everything went pretty smoothly at first, but before long my amateurishness started causing problems. The script became long, badly written, unorganized, and difficult to change. My learning process is now forever entwined in the game script, and because I wrote the core parts of the game first, it's the core parts that are the worst.

In my free time, I'm still working on expanding the game, but the scripting process is slow and painful. By investing a bit of time learning some programming before starting the game, I could have made my life a lot easier.

**2 UNORGANIZED ASSETS.** I had no idea how many hundreds of files I would end up generating for this game when all the different working versions of animation frames, sound effects, levels, music tracks, and menu graphics—plus the backups, finals, and final finals—were added together. The confusingly named files are still scattered all over my hard drive, and even within AGS my asset lists make no sense.

When I began work on the game, I had a few little sprites and there was no problem remembering what was what. Soon, however, I had to start keeping other windows and programs



Screenshots of early builds of GESUNDHEIT!



open so I could check if the asset I'd refer to in the code was what it was supposed to be. In the script, the enemies would be labeled A or B, 1 or 2, or color-coded depending on my whims, and the level names were messes of numbers, dashes, and underscores. I was constantly telling myself that I should sit down and do a major clean-up, but I perpetually thought I was almost done and that it wouldn't be worth the effort. I was wrong in both cases.

**3 USING 3D SOFTWARE FOR 2D ANIMATION.** I've described how I made my character art with a combination of scratchboard artwork and Photoshop, but for the animation I actually used a 3D program. Why use 3D software for 2D animation?

For some reason, I had become obsessed with the idea that all my characters needed to have really smooth joint bends (for example, knees and elbows) and this would be difficult in the 2D software I knew. But I thought I could make it work in a 3D program using bones. Using Anim8or (a freeware 3D program), I mapped my character art onto flat 2D polygon silhouettes of the characters, as if I were applying paint to a paper cutout. Then

I built skeletons for my characters, animated them (keeping everything flat on the X-Y plane) and rendered out the animation frames from an orthographic camera. Those frames were taken back into

Photoshop where they were resized and formatted for the game, and outputted once more.

Why did I go to all that trouble? Don't ask me! If I could do it again, I would have just done all my animation in After Effects or ImageReady. Going through a 3D program was a huge hassle, and since the final sprites are only something like 30 pixels tall (with elbows only a few pixels wide) the smoothness of the bends is sadly not one of the most notable features of the game.

**4 LATE PLAYTESTING.** Being the shy fellow I am, I was hesitant at first to let strangers play my game before it was done. I didn't mind showing classmates my work in progress, but most of them weren't gamers. Even those who were into video games seemed hesitant to critique the gameplay—they were too kind to hurt my feelings.

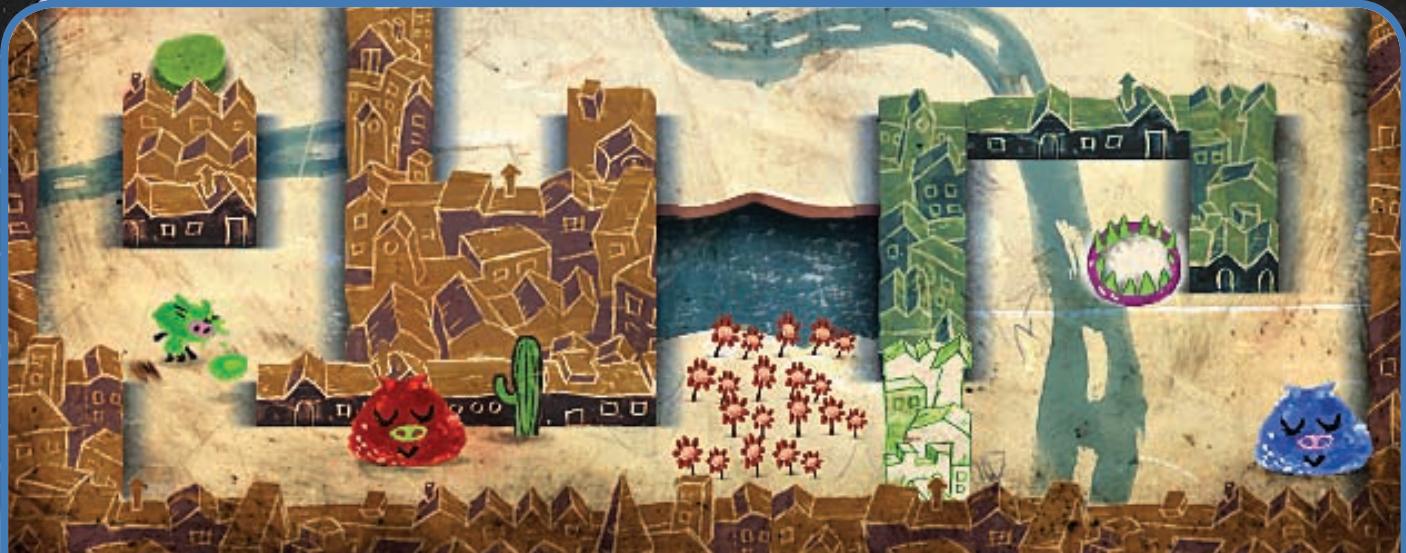
By the time the game was ready for the fresh eyes of play testers, I was so close to being finished that I didn't want to wait and get really thorough feedback. The handful of AGS play testers I had gave great advice, but there were a few big things, such as the lack of keyboard controls or an aiming aid, that drove some people mad once I posted it online. As I was working on the game, I had become really good at playing it, and I didn't even notice how cruel the game was for new players. I should have gotten a broader sample of testers to give feedback earlier in the process.

I was able to add and improve some features in subsequent versions, but if I'd considered these issues early on, they could have been much easier to implement and incorporate into the rest of the game script. And I don't think it would have been hard to find testers—my experience with the indie gaming scene is that everybody's only too willing to offer feedback.

CONTINUED ON PG 44



## POSTMORTEM



CONTINUED FROM PG 43

**5 DIDN'T RESEARCH ENGINE ENOUGH.** At the start, I was so eager to jump right into development that I didn't spend much time researching my engine. The version of AGS I used [2.72] was great, but there are Mac ports for older AGS versions that I could have taken advantage of if I'd looked into it first.

And that's only AGS. I often wonder what the game could have been like if I'd used another engine altogether. Online play? A level editor? When I started the game I never considered anything besides AGS, and realistically, features like that would probably have only bogged me down. Regardless, it would have been nice to have those options for future versions.

### GOOD HEALTH TO YOU!

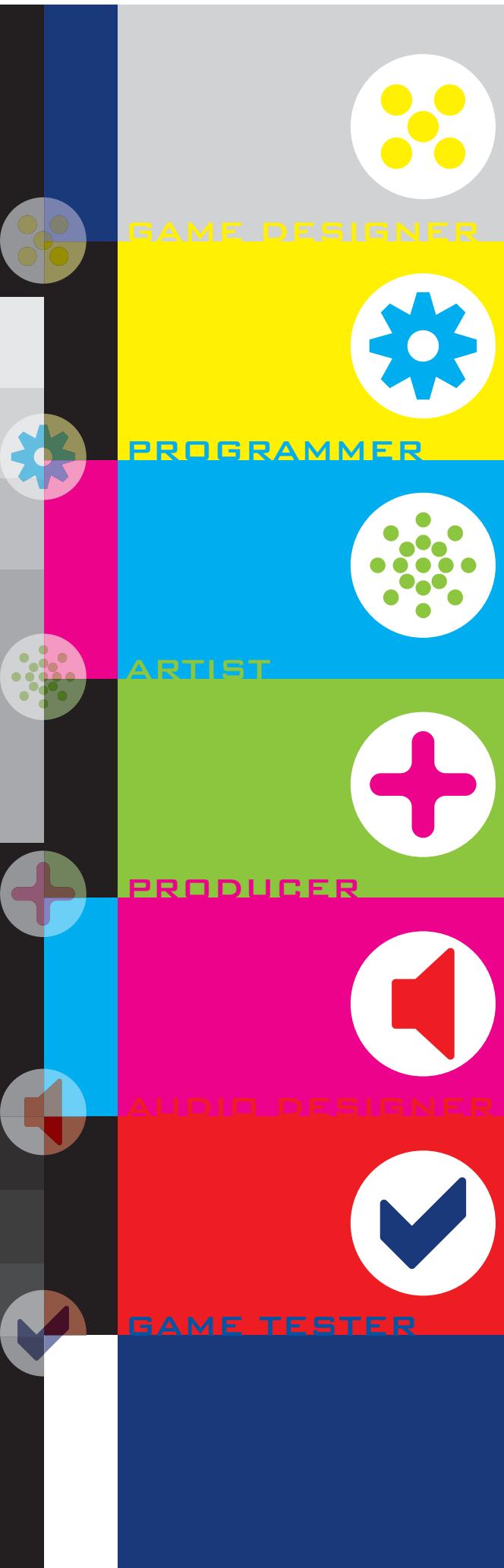
A lot of the problems I encountered while working on GESUNDHEIT! are probably the most common and obvious mistakes a newbie can make, and I'm sure the game could have been done better if I'd had someone with game experience guiding me. Yet,

stumbling through the process was part of the fun, so it's hard to view any of it with regret.

Doing the game for an art course was very liberating, and having an experienced illustrator give me feedback helped me push the graphics much further than I would have on my own. And it was quite a treat at the end of term to show off my own video game to the rest of the class on the school's digital projector.

Making GESUNDHEIT! was probably the most fun I've had on a solo project. Learning stuff is cool, and I did nothing but learn the whole way through. My inner control freak loved working out everything from the scripting to the snot stains, and I'm better at the recorder than I've ever been before. As an aside, GESUNDHEIT! was my first serious attempt at character animation. I ended up enjoying that part so much that I took a postgraduate course in the subject. I start my first job at an animation studio next week, so I guess I owe a lot to my little green pig and his smoothly bending elbows. ☺





» albert t. ferrer

## THE DISCIPLINES

Jobs in video game development can be categorized by discipline. Here we explain what the major disciplines are, what kinds of job titles they hold, and what makes each one appealing to different types of people. For salary information about each discipline, see “Paycheck Please!” on page 11.



### GAME DESIGNER

Game designers typically determine the overall vision of a video game. Much like film directors are to movies, designers are known for having a large influence (creative or otherwise) on the direction a game takes, from the early concept stage to final release. Ideally, a designer should be knowledgeable in different aspects of game development, since the role itself calls for collaboration with various departments: art, programming, production, quality assurance, talent (voice actors), audio, and marketing.

Depending on the size of a game development studio and the needs of the product, designers can take on various roles. Sometimes they do a bit of everything, while other times they are assigned only a few specific tasks. These specialized tasks can include designing levels, working on gameplay, placing enemies in a game, or writing the game's dialogue.

The pecking order among game designers is design director, lead designer, and junior designer—and there are dozens of titles in between those benchmarks.

Much of a game designer's job involves writing documentation, called the game design document. The design document is made up of both text and diagrams, and it conveys different areas of the game in a relatively clear yet technical manner. Its purpose is to give a sense of how the game will work. The

---

**ALBERT T. FERRER** is an artist, animator, and freelance contributing game writer from Vancouver, currently working in the game industry. Email him at [aferrer@gdmag.com](mailto:aferrer@gdmag.com).

# THE DISCIPLINES

design document is similar to a manual, but with much more detail on how each aspect of the production will work and how it all interrelates.

Being a game designer is a complex role. It's more about maintaining, implementing, and executing ideas than solely coming up with a storyline and some characters. Being an avid gamer is simply not enough, which is why a good designer should have a well-rounded education. Creating a game is a collaborative effort of many talents, and oftentimes results in compromises between what the designer wants and what the other departments can provide.



## PROGRAMMER

Game programmers or software engineers work at the coding level to simply make a game work. They're responsible for implementing various features requested by the designers, as well as assets provided by the artists. They patch together the individual pieces of the game into what will hopefully become a fully playable piece of software by the end of the production cycle. Without programmers, game ideas would never be made into functional software.

Programmers are the most in-demand type of game developers and are in the shortest supply; therefore, they tend to be paid very well. Programmers typically choose to study computer science, sometimes at the graduate level, though degrees in the more specific field of game programming are popular as well. Game programmers must be knowledgeable in a range of computer languages, the most important being C++. They should also have a grasp of advanced math and be good problem solvers. Some programmers are self-taught, which is perfectly acceptable in the game industry. Employers place more emphasis on what a prospective programmer can do than on where and how they learned to do it.

The position of programmer in a game studio can be broken down into areas of specialty: gameplay programmer, AI programmer, tools programmer, graphics programmer, to name a few. Having clearly defined roles helps ensure that each aspect of a game has a programmer specifically devoted to it. Some of the job titles a game programmer could hold are junior programmer, lead programmer, senior programmer, technical director, and tools engineer (someone who builds proprietary tools).

Programmers are typically the first ones to come on board for a game project (aside from lead designers), and are often the last ones to leave. They tend to work the most overtime, especially in the final weeks before a game is shipped.

A test is almost always part of the interviewing and hiring process for game programmers. These tests can range from a take-home test, an in-office computer-based exam, or a live "whiteboard" exam, where the interviewee works out a problem on a whiteboard in front of the interviewers.

Programmers work closely with designers to figure out even the smallest details about how a game should work, using design documentation as a guide to begin connecting all the complex pieces of game development into a functional piece of software.

Though code is what makes a game work, it can also create problems which programmers must constantly fix as the game grows with additional functions and assets. Programmers are an integral part of quality assurance. Working with game

testers, programmers are notified of bugs or broken aspects of the game, communicating with testers regularly throughout the development cycle.



## ARTISTS

Game artists bring to the players' eyes the vision set out by the designers, art director, and producers. From the concept artist, who works with the art director to establish the game's style, to the 3D modelers, who realize those concepts, artists play a critical role in breathing life into a game.

Game artists and animators should be familiar with at least one major 3D software application (examples include 3ds Max, Maya, Softimage XSI, and Blender) as well as 2D graphics tools, especially Photoshop. They should have a foundational knowledge of fine arts first, upon which they can build additional digital skills. Some game artists have added experience in web site design, while others have studied hand-drawn animation, and still others are former sculptors and painters. A non-university post secondary education is common for many artists in the industry, and while a bachelor of fine arts degree is not required, it doesn't hurt. Having a broad knowledge of art is seen as an asset.

Occasionally, there is a specialized person in the art department called the technical artist. It's a role that's becoming more and more common as technology becomes more advanced. Technical artists are still considered artists, but they specialize in the hardware and software side of things, and are typically in mid- to senior-level positions. These artists oftentimes use scripting languages to come up with ways to customize the art workflow. They can also deal with lighting, shaders, and particle effects, as they pertain to game engines.

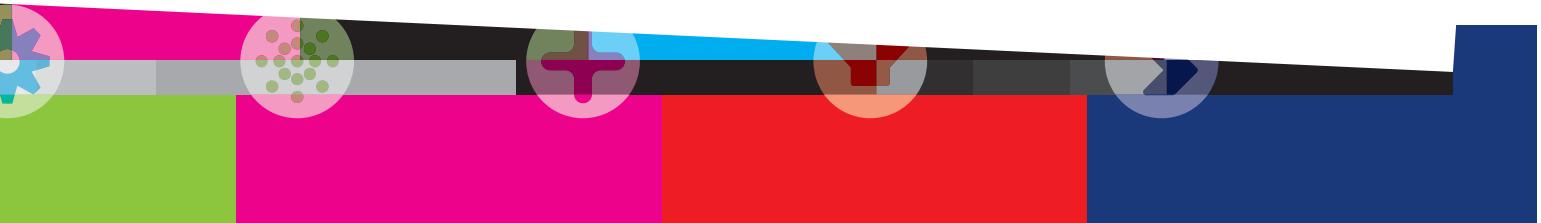
Other titles a game artist might hold include animator, rigger, modeler, user interface artist, character artist, environment artist, concept artist. Many smaller game studios prefer generalists who can work in different roles when required.



## PRODUCER

"Game producer" can be a confusing and somewhat nebulous title. They are essentially the project managers of game making, though the role of project manager is to manage the development team, while the producer deals with high level issues overseeing the project at large. Both fall under the production staff department, and responsibilities for either can vary from studio to studio. Their job is to organize and facilitate the game's production. Producers create and enforce schedules and budgets. They serve as mediators between departments, and sometimes also between the studio and the publisher. They assign tasks, make sure deadlines are adhered to, and generally make sure the team has everything it needs to make the game.

Producers have to be leaders. They absolutely must be able to communicate and get along with a wide variety of people, from gregarious designers to terse programmers, from introverted artists to over-caffeinated testers. They don't necessarily have to be likeable, but being respected by their team is important. They are the point of contact for the lead programmers, lead artists, and directors within the studio.



There are creative producers, who add their design sensibilities and opinions to the benefit of the project, and the not-so creative ones, whose attempts at being creative do more harm than good.

Some of the titles a producer can have include assistant producer, associate producer, and producer, executive producer, and director of production.



## AUDIO DESIGNER

A game audio designer (or sound designer or audio engineer) is responsible for creating the sounds and music to match the visuals of a game. They should be familiar with all the aspects of sound design, from foley to production sound mixing, dialogue editing, and the technology that drives the job, such as Pro Tools and MAX/MSP.

The audio designer's objective is to give the game a unique and distinct sound, like a game's visual style. The job is one part creative aesthetic, and one part technical. Game audio people can also be composers, writing and recording original music for the projects they work on. Ideally, audio designers need a good sense of hearing, adept to the intricacies of sound, and should love music.

There are fewer positions for audio designers than artists, programmers, or designers. A game team may employ dozens of artists and programmers, but only hire one or two audio designers, and even then they are often contract employees. Because they tend to work on contract, audio designers can pursue other musical gigs outside the game industry too; often they take work in film, television, and advertisement.



## QUALITY ASSURANCE TESTER

Considered the last line of defense before a game hits the shelves, game testers help ensure that the quality of the game meets strict guidelines set by the development studio and the publisher. Game testers often work part-time and are paid by the hour. They do not need to hold college degrees—in fact, college and high school students sometimes take testing jobs as part-time work or as a summer job.

People in the department play the game or portions of the game while looking for and recording bugs, glitches, or other major problems. When they find a bug, they test to see if they can repeat it, and if they can, they record their bugs in writing for the programmers or artists to fix later; testers need to have solid communication skills.

Using a complex database, the process of testing a game becomes a well-oiled machine with a lot of back and forth communication between departments.

Tester is an entry-level position but is still seen as a way to get one foot in the door to the game development industry. Testers who prove they have great communication skills and understand the different priorities that come into play while developing a video game can become leads, who then can sometimes move into associate or assistant producer roles.



## OTHER JOBS

*Community managers.* Community managers are involved with the online community aspect of a specific game title. They manage and tend to the concerns of the game players through forums, web sites, email, and chat rooms. The community

manager then relays those concerns to marketing teams, publishers, or the developers, operating as the liaison between the public and the people behind the curtain. This is most often seen when developers release beta test versions of their games for the public to play, and the players in turn send feedback, bugs, or other concerns to improve the game.

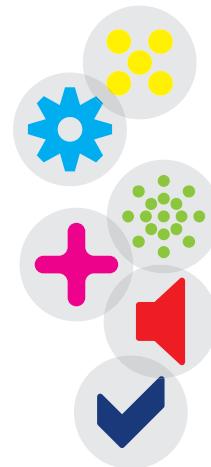
*Play testers.* Play testers are unpaid game testers who play the game much like a focus group. Play testers are different from testers and focus testers in the sense that the information obtained from a play tester is more about the individual's thoughts about a game (rather than seeing their opinions as representing a market share of potential customers). Their job isn't to find bugs, but rather to share their overall thoughts on a game.

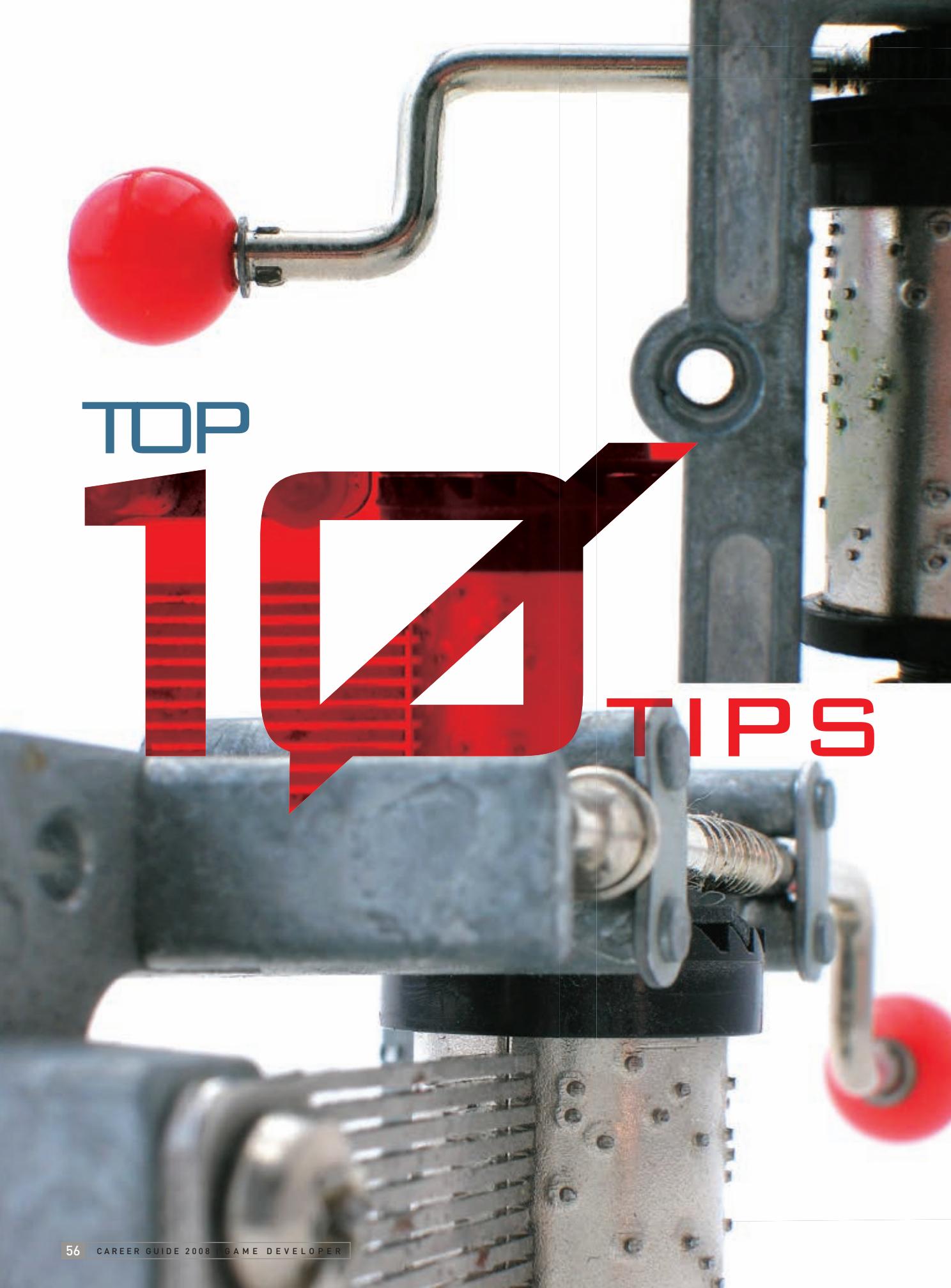
*PR and marketing.* Public relations and marketing people are important to getting a game exposure and into the field of view of the consumer. Marketers use the power of the press, advertising, mailing lists, and web sites to entice players to purchase their games. PR people entice the press to review their games or otherwise mention it in their publications. Getting preview and review copies of games to the media early can help or hinder a game's success. They work with publishers and producers to determine the timing of the marketing, sending out press releases, and launching video teasers, trailers, screenshots, and other assets.

*The press.* Journalists, video game reviewers, the media, or simply "the press" are critical to the success or failure of a game. Their opinions or ratings of a game are highly influential to consumers; they can sway the purchasing decisions of the public. Through the internet or print media, journalists write reviews, previews, commentary, opinion-editorials, and features about video games and the game industry. They are the source the consumer looks to and trusts to get a serious opinion on the latest games. Many game writers get their foot in the door by writing on their own time, either via a blog or on forums, where editors of magazines and professional web sites see their work and make contact with them. A formal education isn't required to become a game journalist, but knowledge of grammar, sentence variation, paragraph and story structure, and copy editing goes a long way.

*Tech support.* Tech support, also known as "production technology" or IT, is the backbone of any game development studio. Within a developer lies a large complicated network of servers and computers that the technical support staff maintains, doing much more than doling out keyboards and computer monitors. They make sure that the valuable information stored within the studio's network is secure. An understanding of networking and programming languages, such as Python, is important. Every game company requires tech support to run a fast network of computers and servers, and to meet the everyday technical needs of employees at game companies.

*HR.* Like any other employer, game studios need human resources professionals. HR staff help manage employees' benefits, protect employees, and typically have to have an understanding of state employer laws. HR staff are also an integral part of recruiting, hiring, firing, and training employees. They keep track of all employee information, and are typically the first people an applicant meets during an interview. Although HR is an administrative role common to all businesses, game companies like to hire people who love video games and know a thing or two about them. ☺





TOP  
**104** TIPS

**1 WRITE A GAME DESIGN DOCUMENT.** Game design documents are at the heart of game design. Every game designer writes some or all of one during development. Since the written word is one of the most important tools in the designer's toolbox, it's worth it to know how to write well. Although there are many different books on the subject of writing, the only real way to get better is to keep writing. Check out Gamasutra.com's series called Anatomy of a Design Document for some more info on writing a GDD.

**2 BUILD A 3D LEVEL.** Building a level is where the rubber hits the road in game design. This is where 90 percent of the game mechanics, game art, inventory systems, scripting, coding, and AI are finally placed. Level building is so fundamental to game design, it's almost impossible to get a job without having one in your portfolio.

**3 WRITE A GAME SCRIPT.** In this case, scripting means coding. There are many good reasons for a game designer to do some scripting. Since designers should always be intimately familiar with the gameplay, having them do the scripting really speeds up iteration of game elements. Plus, it gives designers a good frame of reference when talking to programmers.

**4 MAKE A GAME, OR A MOD.** Getting on a mod project is a great way to learn the team process of building a game. In a game studio's eyes, a designer who has been "in the trenches" is ten times more valuable than one that hasn't. Mod projects are the closest thing to a real development team you can get without actually being paid for it. And, on occasion, mod projects get picked up and published by real game publishers.

Of course, there's no better way to get your feet wet than by trying to do it on your own. There are a bunch of inexpensive game engines and tools out there you can try out, in order to make a full game yourself. Garage Games' Torque Game Builder and Torque Game Engine are two of the most popular.

**5 LEARN TO MOCK THINGS UP IN FLASH.** If a picture is worth 1000 words, a moving picture is worth 10,000 words. Designers love text. It's easy and fast to write down your idea for a game mechanic, level, or system. But is it the best way to communicate your idea? Probably not. George Lucas has used moving storyboards (called animatics) to great effect in the development of the *Star Wars* series. Similar to George's animatics, it's easier for developers to understand an idea if they see it working in Flash rather than reading about it.

**6 BE AN ENABLER.** Game designers are caretakers of a game's vision. As a vision caretaker, you should do whatever it takes to enable the other team members to execute that vision. If an animator asks you to stand up in front of a giant meeting room full of people and dance like a panda to demonstrate an animation to the rest of the art staff, do it! A few minutes of embarrassment is worth it when you ship a great game in the end.

## **7 DISSECT GAMES AND THEIR MECHANICS, EVEN BAD ONES.**

Games are about systems and mechanics. Anyone can come up with an idea for a game. Game designers translate those ideas into core game mechanics and game systems that are fun to play. Being able to critically analyze a game mechanic and figure out why it's so much fun is essential to being a game designer.

I've met a lot of people in the game industry (including designers) who don't play games. We are game designers. We should live and breathe games. It's important to have a wealth of reference so that you can learn what's good and what's bad in game design.

**8 BUILD AN ONLINE PORTFOLIO.** Game designers usually don't have portfolios, but there's a lot to being a game designer that can be posted on the web for prospective employers to look at. Things you might want to have in an online portfolio include writing samples (fiction and technical writing samples), code samples of any scripting or programming you may have done, screenshots of models or levels you have built, and links to any mod projects you might be involved in.

**9 WORK ON YOUR COMMUNICATION SKILLS.** Most of what being a designer is about is communicating to other people the ideas that are in your head. Different people learn things in different ways, so explaining your ideas to other people in their language is only going to help you get your idea across. You might argue that the people you're talking to need to learn to communicate better, but as keeper of the vision of the game you're responsible for making sure that vision is understood by all the members of your team.

## **10 LEARN SOMETHING NEW EVERY DAY.**

You never know what you're going to need to know. When I started my career, I worked on fantasy role-playing games. There was no reason I would expect to need to know about the firearms used in World War II. A few short years later, I ended up working on a WWII shooter, and suddenly, all that useless information was not so useless. And although it helps, game designers don't have to have a degree in game design. All sorts of degrees apply to game design, from English, to History, to Psychology.

**TIM LANG** is lead designer on *TECH DECK LIVE* at Spin Master Studios. Email him at [tlang@gdmag.com](mailto:tlang@gdmag.com).



BY TIM LANG

**1 CHOOSE AN INITIAL TRAJECTORY.** Often the hardest part of breaking into games is choosing where to start. The most traditional path has been through quality assurance, but the number of entry options for the aspiring game producer has increased significantly over the past decade. Bachelor's and Master's degrees in game production are now offered at several colleges and universities, and the number of industry internships has ballooned rapidly. Do not dismiss the prospect of entering games via other disciplines such as art or engineering—it worked for me.

**2 GET YOURSELF OUT THERE.** As a game producer—and this holds especially true for producers with no or limited industry experience—your personality is often your greatest asset. The communication and leadership skills that you bring to a team start with an interview, and the best way to get that interview is with a face-to-face introduction. Even if you are brand new to games, plan on attending industry related activities as often as possible: GDC, local IGDA chapter functions, etc. Put your face and your business card in front of industry professionals, ask questions, and listen!

**3 WHEN IS A PRODUCER NOT A PRODUCER?** Companies differ in their naming practices for game producers: executive producer, senior producer, producer, associate, assistant, and on from there. The responsibilities of each title can change from company to company. Also remember that a producer at a game publisher, which is external from development studios, may have very different responsibilities from those of a development producer, who will be internal to those studios. Do your research and read the job descriptions carefully before you apply for a new production position.

**4 YOU ARE YOUR RESUME AND COVER LETTER!** Unlike applicants to other development disciplines, a qualified producer with little or no industry experience may not have a game demo or portfolio to represent his or her skill-set. One of the keys to locking in that first interview is a well-organized resume, and well written and creative cover letter. As Marc Mencher so aptly puts it, "If you can't organize a simple resume, how could you possibly organize the production of a game?" Research what makes a resume stand out and capitalize on it, and be prepared to discuss your listed experience in-depth during the interview.

**5 MASTER YOUR BUSINESS APPLICATIONS.** Any producer in the game industry will tell you that the tools they use most often are not Visual Studio or Maya, but Microsoft Office and Outlook! As a game producer you are not only required to know these business tools, but to master them. Be prepared to go deep with your understanding of Excel macros and templates, and pick up and learn MS Project. A great team needs a solid schedule, and good schedules can be made quickly and accurately by understanding these tools in depth.



**6 EXPAND YOUR KNOWLEDGE OF OTHER DISCIPLINES.** Pick up the artists' tools and try working with them; play the games your designers are referencing, and learn some scripting or C++. Artists, designers and engineers each have a unique language and culture specific to their disciplines: the effective producer learns to navigate and communicate effectively in each one.

**7 BECOME A NEXUS OF CREATIVE COLLABORATION.** Teams are an eclectic mix of artists, engineers and designers—and that mix doesn't always get along! Production is often the final arbiter of a project's direction, so do not wait until the last minute to build consensus on important decisions. Identify areas of conflict early, and proactively seek out all sides in finding the best solution. The more you dive into conflict and shift the team toward resolution, the more likely they will do it instinctively the next time around.

**8 LEARN TO KILL YOUR SACRED COWS.** Effective production is a balance of time, money and quality. Everyone contributes great ideas, but not all great ideas can make it into the final product, including your own pet projects or features. When it comes to maintaining scope and schedule be prepared to not only cut your teammates' ideas, but also your own.

**9 BE AN EFFECTIVE MANAGER AND A DEDICATED LEADER.** Management is good scheduling, peerless organization, and effective prioritization. Leadership is thorough and constant communication, conflict resolution, and team building. Good game producers excel at one or the other; great game producers are both effective leaders and meticulous managers. Learn to understand the difference, and remember that teams need a balance of both to succeed.

**10 BE PREPARED FOR CHANGE.** A career in game production can be an exciting and rewarding one, but only if you put effort into it! Learn what management or your clients need from your projects and deliver on it. Look for new opportunities at similar companies when you find your career hitting a wall. Research new methodologies like Scrum and extreme programming and be prepared to introduce them to your teams where needed and applicable. Every team you manage will be different and will require a slightly different approach, so be prepared to affect change every day.

Game production is not for the shy or faint of heart: it is ultimately the game producer's responsibility to ensure that their team creates a quality title within the confines of the budget and schedule, and with each passing year more games are green-lit with millions of dollars, hundreds of employees, and years of development on the line. Though they shoulder a great burden for the team, in the end the satisfaction that comes from a project run well and a game well made makes the producer's success all the more satisfying.

---

**MICHAEL J. BOCCIERI** is a producer at Secret Level, a Sega studio.  
Email him at [mboccieri@gdmag.com](mailto:mboccieri@gdmag.com).

BY MICHAEL J. BOCCIERI



BY NOEL LLOPIS

**1 WRITE GAMES.** This is the most important thing any aspiring game programmer can do. It doesn't matter what language you use, or what type of game it is, or even whether it's very fun. What's important is that you go through the experience of writing several full games from start to end and get a feel for how everything works together. I recommend working on several smaller games rather than one huge one. Writing a mod is good, but make sure to write a few with your own, simple technology. These games will make great demos to strengthen your resume.

**2 COVER THE BASICS.** As a game programmer, there are certain techniques you'll keep reaching for in your bag of tools. Make sure you get comfortable with matrices, vectors, and their operations; with basic data structures and algorithms; and with the fundamentals of software engineering and object-oriented programming. Getting a good foundation in those areas will allow you to concentrate on creating a game and not fumbling with basic concepts.

**3 ACHIEVE INTERMEDIATE LEVEL IN C++.** Most games are still programmed in C++, and to be an effective team member, you'll need at least an intermediate level of C++. Don't worry about mastering the language yet. C++ is full of dusty, dark corners that only language lawyers know about. Concentrate on writing solid, clean code without language trickery. Don't obsess over specific APIs (DirectX, OpenGL, etc.). Chances are you'll use a different API or that it will be wrapped in a higher-level interface.

Bonus points: Learn another, very different language. Python, Lua, or Lisp are good candidates. They'll give you a whole new perspective on programming and make you a more effective programmer all around, and, depending on the project, they might even come in handy for game development.

**4 NEVER STOP LEARNING.** Don't think for a moment that as soon as you're done in school you will have learned everything you need. The best game programmers are always actively learning new things, even after many years of experience. There's no shortage of new things to learn about: new hardware, languages, techniques, APIs, algorithms, and more. And with so many resources on game development out there, there's no excuse not to be constantly learning from books, magazines, web sites, and conferences.

**5 COMMUNICATE.** I can't emphasize enough how important good communication is when working in a team. You'll have to communicate effectively to coordinate efforts with other programmers, to explain how your code works to artists and designers, to understand what features the rest of the team needs, and to explain to your boss what you've been doing and offer suggestions on how to make it better. How do you become a better communicator? Give presentations in your class, share

what you've been doing with other people over lunch, write a blog detailing your experiences, or participate in forums and mailing lists on game development.

**6 TOOLS OF THE TRADE.** Learning all the theory about programming and algorithms is great, but when it comes down to it, you'll be using certain programming tools all day long. Get comfortable using a good editor, a source control program, and a debugger. Don't worry whether they're the same ones you'll be using in your job or not. The important thing is to know what you can do with them and how to use them effectively.

**7 WORK WITH A TEAM ON A LARGE PROJECT.** If you only work on projects by yourself, you're going to be in for a shock when you land your first job in the games industry. Not only will you have to coordinate how your code works with other people, but you'll have to read and modify large amounts of code written by other members of the team as well. If you don't have the chance to work on large projects for your courses, join an existing open-source project to get some of the same experience.

**8 PLAY AND STUDY GAMES.** You should be familiar with the latest games out there, and with the latest advances in game technology. Don't just play the games, but study them. Try to figure out what shadow rendering technique they're using, or how their AI is making those tactical decisions. Find out more information through postmortems or mailing lists and think about if those techniques could be applied to your games.

**9 PICK AN AREA OF EXPERTISE.** An effective game programmer has a very good understanding of how all the different areas of a game work together: graphics, AI, collision detection, networking, user interface, input, physics, and all the rest. At the beginning of your career, it's very important to get exposed to as many parts of the game as possible. At the same time, start thinking about which areas appeal the most to you and focus on those a bit more. After a few years you'll be able to specialize more and become an expert in a given area.

**10 GET A WELL-ROUNDED EDUCATION.** Don't be afraid to branch out and learn things not directly related to programming. Having a well-rounded education or even hobbies totally unrelated to game development can provide a useful foundation that will help you all throughout your career in lots of unexpected ways.

Above all, apply yourself and finish your degree the best you can. Not only will you get a lot out of it, but you'll also show potential employers you can stick to a long-term project and do well in it.

---

**NOEL LLOPIS** threw caution to the wind, gave up a steady paycheck, and decided to follow his lifelong dream of being an indie game developer. He keeps busy by pretending to do everything from programming and design to business and IT at Power of Two Games. Back when he was still getting paid, he worked on *THE BOURNE CONSPIRACY*, *DARKWATCH*, and the *MECHASSAULT* series.

Email him at [nllopis@gdmag.com](mailto:nllopis@gdmag.com).

**1 IN-HOUSE OR FREELANCE?** One of the most fundamental things to decide is whether or not you are looking for a full-time salaried position within a game development company, or if you are more comfortable with offering your audio chops to the game industry as a freelancer. Composers usually fair better in the freelance realm than an in-house situation, where they would be expected to do more than just compose. If you are a talented all-rounder, you may be equally in demand for in-house or freelance positions. The decision may come down to a work-life balance. Once you know what you are looking for, you can more effectively target employers or clients.

**2 ALWAYS TREAT YOUR CLIENTS WITH RESPECT.** Whether you are in-house or freelance, the people you work with should be treated as your clients. As a sound designer, composer or sound implementer, you need your client as much as they need you. They may sometimes come up with suggestions that sound crazy, but listen to their ideas, explore them, work on a few examples and try those suggestions out. You may be surprised—something that sounds crazy at first might just work. As a result, the people you work with will feel included in the creative process and you will be happy with a job well done.

**3 CREATE A SOLID DEMO REEL.** The demo reel is perhaps the single most important piece of work you will present to your prospective employer. So much info about your work and communication style will be communicated through how your reel is edited, structured, and presented. Don't send out the same general reel to lots of companies—it's much better if you can tailor specific footage or examples to a particular company. Include a cover letter explaining specifically why you are interested in that position or company. Keep it simple, clean and always focus on your best work. Also, if you worked on a specific area in a clip of game-play or a movie, such as only the helicopter sounds in a game, make this unequivocally clear at the outset.

**4 MAKE CONNECTIONS AND CONTACTS ALREADY IN THE INDUSTRY.** There are many platforms for this kind of interaction available to people entering the industry, such as GDC. Meeting and chatting with audio talent that is already established in the industry is a great way to make a connection and get some feedback to better hone your job seeking talents. As ambassadors for their companies and for audio in general, people who are presenting lectures, round tables and workshops at conferences are great and approachable contacts to make. Everyone who is successful in game audio now was where you are now at some point in their past.

**5 SUPPLY AND DEMAND.** There is currently a huge surplus of composers in video game sound. Look into an area where there is a shortage. Currently, audio programmers, sound effects designers, sound implementers, dialogue designers are all in much shorter supply than composers, so it makes sense that you are more likely to find ways into the industry via these fields. Once inside the games industry you will get ample chance to prove your talent and move into a role with which you are more comfortable.

**6 LOOK FOR WAYS TO PROVE YOU ARE A TEAM PLAYER.** Game development is about collaboration and about finding mutually creative solutions to problems to better serve the end product. If you can prove that you have worked with an animator or a director on a short film or game, and demonstrate some of the areas where you have talked about sound and worked at integrating other people's ideas into the sound, this will definitely impress whoever is conducting the interview. Great audio is about supporting the gameplay or the story of a game, not about good sound for good sound's sake. If the game fails, everyone fails, whether the audio is good or not.

**7 DON'T EMBELLISH YOUR RESUME.** Putting false or misleading information on a resume or demo reel isn't a good idea, and you will get found out. Always specify exactly what you did on any particular film or game title. If you just did dialogue editing, then just put dialogue editing—don't make out like you did all the audio on the game. You will be asked to talk about your work on any particular title, and employers check references very diligently—no matter how much experience the candidate appears to have.

**8 GET EXPERIENCED.** Employers are usually looking for experience above qualifications. A lot of game audio or film audio courses are very vocational in their approach, so this will count to an employer as production experience—even better if you can approach a game developer for an internship.

**9 EVERYONE IS A CONTACT.** Making contacts can start as early as college. If you can work for free on a friend's project, do it. That friend will most likely get a job or even start a company and go back to you for the sound, or be able to recommend you to others. Again, if you have a positive collaborative experience working with particular people they will remember you and recommend you further down the line.

**10 FLEXIBLE SOFTWARE SKILLS.** Tools for implementing sound in games are always changing, and there is no industry standard for software in the games industry as there is in film. Not everyone uses ProTools. Learn to be competent on several different systems, sequencers, and editors. Learn as wide a range of sequencing software as you can and experiment with projects in Audiokinetic Wwise to get an idea how game audio will be implemented.

**ROB BRIDGETT** is a freelance sound designer. His work for games includes sound design for Dreamcast title *VANISHING POINT*, followed by a 2 year in-house stint at Climax in the UK, and a position as sound director on *SCARFACE: THE WORLD IS YOURS* at Radical Entertainment. Email him at [rbridgett@gdmag.com](mailto:rbridgett@gdmag.com).



BY ROB BRIDGETT

**1 NEVER TURN IN WORK THAT HAS TECHNICAL MISTAKES.** As a professional artist, your job is to give life to what exists only in the imagination, and other people rely on you to do this with a certain level of proficiency. When you turn in work that has obvious anatomy mistakes, incorrect perspective, or contradicting lighting, you are basically telling your co-workers that you are an incompetent artist. Ideally, your work should contain as few technical mistakes as possible, and the only critiques you get from your peers and superiors should regard style and design, never your competence as an artist.

**2 NEVER TURN IN SLOPPY WORK.** As an art director, I constantly see sloppy mistakes which are just inexcusable—white halos around cropped images, jaggies around edges, stray pixels in alpha channels, reversed normals, double faces on a polygon, messed up UV, and plenty more. As a professional production artist, you should be turning in work that is clean, up to spec, and ready to run in the game. Take that extra 10 minutes to check your art asset on a local copy of the game to make sure it works before officially checking it in on the server. Don't be the one who breaks the latest build with your sloppy art asset.

**3 ALWAYS GET APPROVAL BETWEEN STAGES.** Often artists will work in the dark and only realize they've been going in the wrong direction when they resurface with a new batch of work. The art director won't always have time to come and check on your progress, so you need to make sure you are getting approval during each agreed-upon stage.

**4 DON'T BE NEEDY.** The reverse is also true—don't be insecure and needy. If your task is to do concepts for 10 different combat vehicles, don't go and bug the art director after you sketch each one. Ask the AD at what stages he or she would like you to submit assets for review, and how many variations you should be creating.

**5 ALWAYS SAVE IN ITERATIONS.** Often as we iterate and change a piece of art, we get further and further away from the original intent—sometimes in a negative way. Save in iterations and compare what you have to the earlier versions, and you might be surprised to find that you actually prefer an earlier version more. Saving in iterations means you can always retrace your steps and revert to an older version if necessary.



BY ROBERT CHANG

**6 ALWAYS USE REFERENCES WHEN NECESSARY.** If you have any baggage about using references for your artwork, get over it immediately. Professional artists are required to use references all the time, as there is no better way to achieve authenticity and accuracy. Human brains are simply not that amazing when it comes to recalling the detail of an overwhelming number of objects, animals, buildings, cars, or anything else. Even if you are working on a totally fictional world, you can still reference the real world for inspiration and ideas.

**7 LEARN SHORTCUTS.** Shortcuts can mean literally the ones used in software, or simply ways of doing things. When you know all the important shortcuts in your most-used software, you can dramatically speed up your workflow. For example, Photoshop Actions can save you a ton of wasted time if you create them to perform the repetitive multi-step actions you use often. Shortcut keys also allow you to zoom through many tools and settings without ever having to find anything in the menu or tool bar.

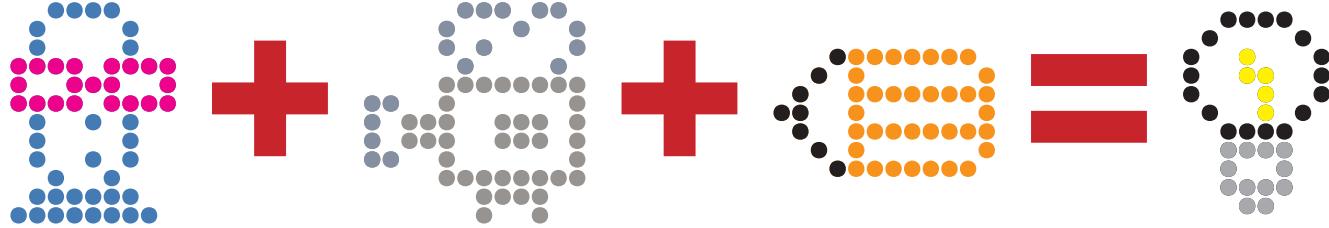
**8 BRUSH UP ON YOUR FOUNDATIONS.** Many artists get their first jobs before they have truly mastered the important foundations they need to know as professional artists. From that first job on, they just tread in the same spot and virtually stop growing and learning as artists aside from what they do at work. Don't be one of them. Keep on learning and strengthen your weaknesses, learn new tools, and try new methods.

**9 DON'T BE A PRODUCTION MACHINE.** As a creative talent, you are hired for your creativity, not just your technical skills. Don't just do what you're told and nothing more—be creative, come up with ideas, and suggest alternative approaches. Artists who just go through the motion with no sense of artistic pride or sense of ownership are typically called hacks.

**10 DON'T BE A PRIMA DONNA.** At the same time, don't be a difficult person. Realize that although you have your own artistic style and personal taste, as a production member, you have to work well with others and be able to follow orders. If you disagree with every art direction, refuse to make changes, and march to your own beat regardless of what's asked of you, then don't be surprised if you're the next person to be let go when it's time for a layoff [if you don't get fired first].

**ROBERT CHANG** just finished up a stint as studio art director at iWin. He is also the co-author of *d'artiste: Digital Painting*. Email him at [rchang@gdmag.com](mailto:rchang@gdmag.com).

# GAME DEV 101



## THE GAME DESIGNERS' VIDEO GAME QUIZ



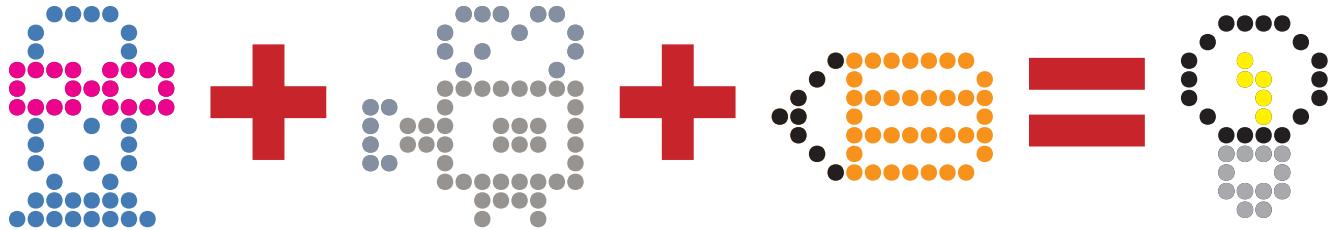
HOW MUCH DO YOU KNOW ABOUT THE GAME INDUSTRY AND ITS HISTORY?



Which of the above designers has not had his name on a game box? Pictured clockwise from top left are Bill Budge, Sid Meier, David Jaffe, and American McGee.

**THOMAS GROVÉ** is a freelance game designer and entrepreneur residing in San Francisco. He is currently exploring collaborative opportunities as co-production company Studio Interrupt. Email him at [tgrove@gdmag.com](mailto:tgrove@gdmag.com).

- » 1] When making a PlayStation series action game, which button do most designers put "jump" on?
  - A – Triangle
  - B – Square
  - C – X
  - D – Circle
  
- 2] SHADOW OF THE COLOSSUS designer Fumito Ueda wanted players to feel apprehensive about jumping. What button did he assign "jump" to?
  - A – Triangle
  - B – Square
  - C – X
  - D – Circle
  
- 3] Which of the following gaming "platforms" has the largest install base as of 2008?
  - A – PlayStation 3
  - B – Xbox 360
  - C – Wii
  - D – Flash
  
- 4] Which of the following is not a production methodology?
  - A – Scrum
  - B – Waterfall
  - C – The Cerny Method
  - D – Counterpoint
  
- 5] Which of the following are present in most 2D and 3D games?
  - A – Polygons
  - B – Pixels
  - C – Tile Maps
  - D – Vertex Lighting



6] Sega was founded in which country?

- A – Japan
- B – Korea
- C – United States
- D – France



7] Which of the following companies commands a force of 1,000 or more game developers?

- A – Electronic Arts
- B – Ubisoft
- C – Tose
- D – They all do!



8] Which of the following terms used in game development is not borrowed from film terminology?

- A – Level of detail
- B – Depth of field
- C – Motion blur
- D – Anamorphic



9] "Rock, Paper, Scissors" is often used in game design for what?

- A – Choosing which movie tie-in to work on next
- B – Balancing different factions
- C – Providing a framework for strategy
- D – Beating bosses in ALEX KIDD



10] Which of the following is not a development platform for mobile games?

- A – BREW
- B – J2ME
- C – Android
- D – GX12



11] Mario is what type of game genre?

- A – Platformer
- B – Cart Racer
- C – Sports
- D – Mario is a character, not a genre.



12] Which of the following is not in the "adventure game" genre?

- A – KING'S QUEST
- B – DARK CLOUD
- C – SAM & MAX
- D – HOTEL DUSK



13] Speaking of SAM & MAX, what has Telltale Games pioneered with that series?

- A – Episodic content
- B – Digital distribution
- C – Digital rights management
- D – Web deployment



14] Which of the following game directors has not had his name on the game box?

- A – Sid Meier
- B – David Jaffe
- C – American McGee
- D – Bill Budge



15] Which of the following game developers has been inducted into the French knighthood?

- A – Shigeru Miyamoto
- B – Michel Ancel
- C – Peter Molyneux
- D – All of the above



16] In what country were game microtransactions popularized and refined?

- A – Japan
- B – Korea
- C – United States
- D – France



17] Steven Spielberg collaborated with EA to make which of the following games?

- A – CALL OF DUTY
- B – MEDAL OF HONOR
- C – BOOM BLOX
- D – ARMY OF TWO



18] Which "scene" made use of the techniques procedural content generation and bloom lighting before either was adopted in mainstream video games?

- A – Demo
- B – Rave
- C – Tracker
- D – Ansi



19] Match the following MMOs with their front-end technology:

- |                       |               |
|-----------------------|---------------|
| I – PUZZLE PIRATES    | A – Flash     |
| II – DOFUS            | B – Shockwave |
| III – SHERWOOD        | C – Java      |
| IV – MINIONS OF MIRTH | D – Torque    |



20] Short answer: What is ironic about 3D Realms forthcoming game DUKE NUKEM FOREVER?

Answer Key

1:C, 2:A, 3:D, 4:D, 5:B, 6:C, 7:D, 8:A, 9: C, 10:D, 11:D, 12:B, 13:A, 14:B, 15:D, 16:B, 17:C, 18:A, 19: I:C, II:A, III:B, IV:D, 20:The game has been in development for what seems like forever.

# FIVE WAYS TO GET INTO THE GAME INDUSTRY

**SO YOU WANT TO MAKE VIDEO GAMES FOR** a living! Luckily for you, this is a great time to be an aspiring game developer. This growing industry is always looking for fresh souls to crush under the millstone of despair and broken dreams!

Of course, all the enthusiasm in the world won't get you a job on its own. You'll need to pay your dues one way or another before you can crash the never-ending party that is the world of modern game development. That initial step often vexes the uninitiated, so let's take a look at some various ways an aspirant can get into this exclusive little club.

## JUST APPLY OUT OF THE BLUE

If you're a gifted artist or a brilliant programmer, you can simply apply on the strength of what you say you can do; no portfolio or remarkable demo required. Just send in a cover letter noting that you "really love games." Surely, no other job candidate loves games as much as you do, and surely none of them have professed their love in a cover letter, as you have. Your resume will totally stand out.

Take note that there are a lot of studios out there, each trying its best to convince you—the talent!—that its office is where the hip and with-it people are, doing exciting and interesting things. But don't just take their word for it! The job ad they post will often evince the truth hidden behind those wooden smiles. Some key phrases to watch out for are:

"AAA game project." Read: crunch, pain, and years off your life ... or, a game about

the people who come out and tow your car when it breaks down. Either way, avoid.

"Just minutes from the downtown nightlife!" Who are you kidding? You're never going to drive into town.

"Competitive salaries and benefits." Just because they can compete doesn't mean they can win.

"We are located in a beautiful, natural area with a low cost of living." I hope you like bartering for molasses and kerosene at the general store, hick.

"A team that respects quality of life." Read: We make dumb, boring casual games and trundle home at five to see our families.

"We have free sodas, meatball Fridays, etc." What they don't tell you is that every time someone eats a free meatball, it comes out of your potential future royalty bonus.

"A positive, creative environment." Read: DESIGNER: Hey, why don't we do something really different?

EXEC: No.  
DESIGNER: Okay.

## GET AN INTERNSHIP

More and more, studios are trying to take internships seriously. I say "trying" because you should only take one if you can stand the constant ridicule of the salaried employees around you. In return, you'll receive a parody of remuneration (if that) and the character-building experience of whatever menial labor they make you perform for their amusement.

Of course, if you think your Master's degree would actually be put to better use if you hauled a thousand fast food tacos in the trunk of your own car to a frustrated and irritable crowd of "AAA" game developers on a perfectly beautiful Sunday afternoon, hey, don't let my cynicism stop you.

## JOIN THE MOD COMMUNITY

Before there were special game schools,

there was the mod community, a rambunctious scene from which many of the industry's best designers can trace their origin. Creating maps on your own is one way to do it, although collaborating in a group via the InterWebs is a great way to prepare yourself to deal with the incessant politics, high drama, and staggering flakiness that you'll encounter once you go pro.

## BE A SMARTASS WHO WRITES ABOUT GAMES ON THE INTERNET

There are multitudes more smarter smartasses who write about games on the Internet than there are openings at game companies for such folks. If you take this route, you'll be your own boss, and your salary will reflect how many "male growth enhancement" banner ads you can sell on your homepage. Good luck!

## THE PATH OF THE TESTER

You may be surprised to learn that a not-insignificant number of today's most successful executive producers, creative directors, and designers got their start many years ago in the unglamorous department of testing. Like the pawn that metamorphoses into a powerful queen after traversing the entire chessboard, this transformation is made all the more dramatic for the unlikely odds of its occurrence. Though the path is open to all, the only ones who really have a chance are those who possess a Frodo Baggins-like level of tolerance for travailing against unfathomably vast enemy resources.

And be careful—the fire of adversity sometimes forges a hardened spirit, but other times it simply burns away the soft material of the soul, leaving only a dry husk clattering empty in the wind.

Alternately, just try to be buddies with a producer. ☺

**MATTHEW WASTELAND** is a pseudonymous game developer who has a fairly common first name. Email him at [mwasteland@gdmag.com](mailto:mwasteland@gdmag.com).