

JEDE AUSGABE MIT FIRMENREGISTER

04/2014 € 6,90

OFFIZIELLER PARTNER VON
game
Bundesverband der Computerspielindustrie

making games

DESIGN | BUSINESS | ART | TECHNOLOGY



INNOVATION MANAGEMENT

WIE BLUE BYTE, DEEP SILVER FISHLABS UND BIOWARE GUTE IDEEN
NICHT NUR FÖRDERN, SONDERN AUCH ERFOLGREICH UMSETZEN KONNTEN

UNREAL & UNITY
DIE MACHER DER WICHTIGSTEN
GAME ENGINES IM INTERVIEW

GAME DESIGN ANALYSE
WAS CASUAL GAMES VON ONLINE-
ROLLENSPIELEN LERNEN KÖNNEN

LORDS OF THE FALLEN
SO ENTSTEHEN DIE VISUELLEN
EFFEKTE DES NEXTGEN-RPGS

Wir stellen ein – bewirb dich jetzt!

**Ausbildung
Fachinformatiker (m/w)**

**Ausbildung
Büromanagement (m/w)**

**Ausbildung
Mediengestalter (m/w)**

**Technical Lead
Unity 2D/3D (m/w)**

**App Store Marketing
(m/w)**

**Level Designer
(m/w)**

**Unity 2D/3D
Developer (m/w)**

**Game Programmer
(m/w)**

**Game Designer
(m/w)**

**Game Artist
(m/w)**

**Marketing Designer
(m/w)**

**Praktikanten
(m/w)**

Mehr Infos findest Du unter:

www.handy-games.com

www.facebook.com/handygames

www.twitter.com/handy_games



**Praktika
Ausbildung
Studium
Jobs**

JETZT WIRD'S PERSÖNLICH

Es gibt Dinge, die ich in neun Jahren Making Games noch nie gemacht habe. Das Editorial aus der Ich-Perspektive schreiben, zum Beispiel. Schließlich gibt es wohl kaum ein anderes Magazin, das so sehr eine Teamleistung ist wie

Making Games – mittlerweile haben schon weit über 200 unterschiedliche Autoren für uns geschrieben. Da muss es schon verdammt gute Gründe geben, ausnahmsweise mal auf das »Wir« zu verzichten.

Grund 1: Meine Wenigkeit (nicht Making Games!) ist aus der Jury des Deutschen Computerspielpreises ausgetreten. Weil ich mich nach 5 Jahren ehrenamtlicher Tätigkeit nicht mehr mit diesem Preis und vor allem seinem Regelwerk identifizieren konnte. Den Rücktritt haben mein Kollege Andre Peschke und ich in einem Video ausführlich begründet und öffentlich gemacht: www.makinggames.de/dcp-austritt. Was mir dabei wichtig ist: Das war eine persönliche Entscheidung, kein politisches Statement von Making Games. Unser Magazin war schon immer ein unpolitisches Magazin und wird das auch in Zukunft bleiben. Entsprechend werden Sie in diesem Heft von mir weder einen Kommentar zum DCP finden, noch zum kürzlich erfolgten BIU-Beitritt einiger großer Online-Spielefirmen. Das überlasse ich lieber denjenigen, die sich im Detail damit auskennen – im Falle des DCP ist das G.A.M.E.-Geschäftsführer Thorsten

Unger (S. 11), zum Thema BIU-Beitritt äußert sich Gameforge-Prokurist Tobias Haar (S. 7).

Teilzeit-Abschied

Grund 2: Zum 1. Mai habe ich die Chefredaktion der GamePro übernommen, unserer Konsolenspiele-Schwesterzeitschrift bei IDG. Logischerweise bin ich daraufhin vielfach gefragt worden, welche Auswirkungen das auf Making Games haben wird. Die Antwort halten Sie gerade in Ihren Händen: keine. Ich bleibe trotz der GamePro auch hier weiter an Bord. Naturgemäß wird sich mein Zeitbudget für Making Games aufgrund der Doppel-Chefredaktion reduzieren. Aber glücklicherweise habe ich mit Patricia Geiger und Yassin Chakhchoukh ebenso kompetente wie tatkräftige Kollegen. Zum 1. Juli bekommt Making Games außerdem hochkarätige Verstärkung in Person von Sebastian Weber, der unter anderem bereits als Redakteur bei Computec sowie als Localisation Manager bei Kalypso gearbeitet hat. Gemeinsam mit ihm und natürlich auch gemeinsam mit Ihnen haben wir noch einiges vor. Nicht nur diese Ausgabe mit höchst unterschiedlichen, höchst interessanten Artikeln zum Thema »Innovation Management« (ab S. 12). Sondern auch in Zukunft. Als Wir, nicht als Ich.

Viel Spaß mit der aktuellen Ausgabe!

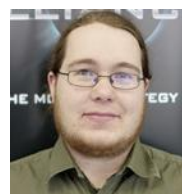
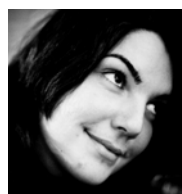
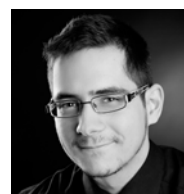
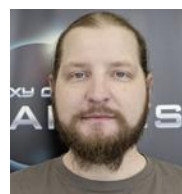
das Team von Making Games

Helko Klinge
Ist Chefredakteur vom
Making Games Magazin.



»Unser Magazin
war schon immer
ein unpolitisches
Magazin.«

Sebastian Weber
Verstärkung für das Team
von Making Games.



INHALT

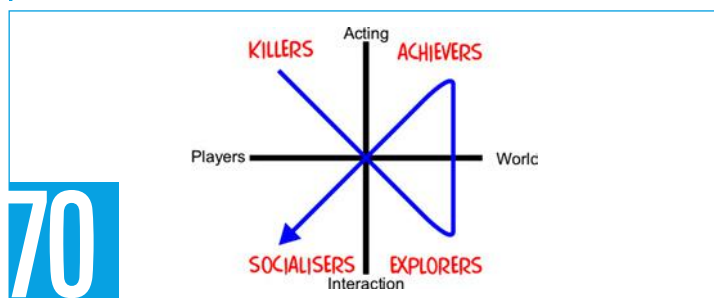
04/2014

- 03** Editorial
- 80** Ein Tag bei...
Black Forest Games
- 82** Vorschau
- 82** Impressum

- 06** Branche / Köpfe
Bemerkenswertes aus der Branche
- 07** Eventkalender
Wichtige Branchenveranstaltungen
- 08** Passiert auf makinggames.de
Interessantes aus dem World Wide Web
- 10** Making Games Tools
Die Meinung der Profis zu aktueller
Soft- und Hardware
- 11** Fokus G.A.M.E.
von Thorsten Unger



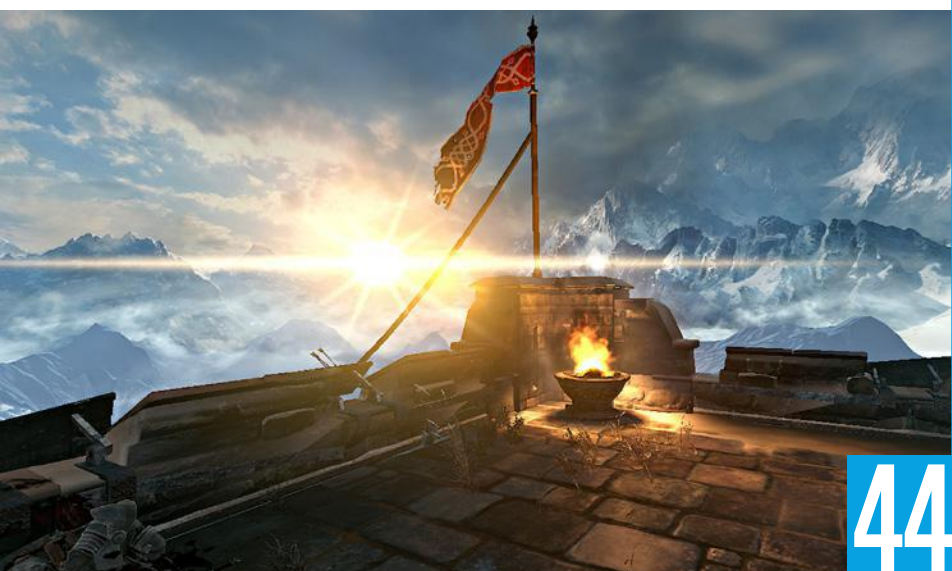
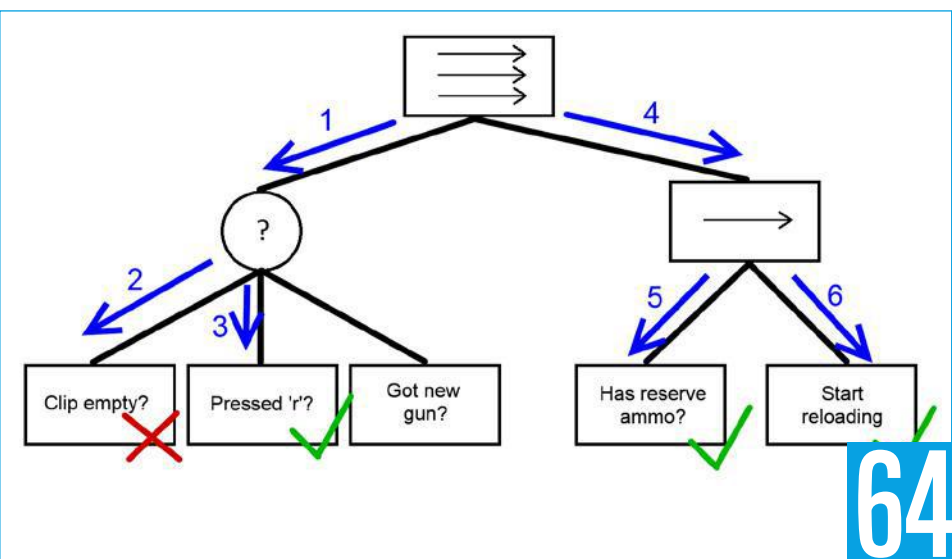
80



TITELSTORY

Innovation Management

- 12** **Essay**
Das ändert alles
von Christian Schmidt
- 16** **Case Study**
Innovating with Flash
von Burhard Ratheiser
- 22** **Post Mortem**
Spoiler Alert – Uncompleting a Game
von Jeff Jensen
- 26** **Case Study**
QA for Innovations at Bioware
von Tulay McNally
- 32** **Case Study**
Ds Backend-Development von
Galaxy on Fire – Alliances
von Nils Plachke und Hans Ulrich Sesselmann



- 40** **Interview mit Andrew Sheppard**
Kabam – von Facebook zu Mobile
von Patricia Geiger
- 44** **Case Study**
Visuelle Effekte für die nächste Generation
von Benjamin Glatzel und David Reinig
- 52** **Post Mortem**
Royal Revolt – die Entwicklung einer neuen
Marke im Mobile Gaming
von Klaas Kersting
- 56** **Interviews mit James Golding und David Helgason**
Die neue Engine-Generation – Teil 1
von Heiko Klinge
- 64** **Forschung**
Behaviour Trees in Cosmonautica
von Philipp Erler
- 70** **Analysis**
Developing Players
von Richard Bartle
- 74** **Firmenregister**
Die Branche im Überblick

E3 2014

DIE SPIELE ENTSCHEIDEN

Das Schlimmste hatte der neue Xbox-Chef Phil Spencer schon hinter sich, noch bevor er die große E3-Bühne betrat. Denn schlimmer als 2013 konnte es für Microsoft schließlich nicht mehr kommen. Aus der einstigen Multimedia-Zentrale mit voll integrierter Kinect und permanenter Online-Anbindung ist nach mehreren Rückwärtssalti seitens Microsoft mittlerweile eine ganz normale Spielekonsole geworden. Entsprechend games-lastig war auch die Pressekonferenz. Kein einziges Wort über TV, Kinect oder auch nur das neue Kinect-lose Xbox-One-Modell. Stattdessen jagte eine Spielvorstellung die andere, allerdings ohne echten System Seller – Halo 5 wurde schließlich schon letztes Jahr enthüllt, erscheint aber erst 2015.

Den System Seller hatte Sony zwar mit der erwarteten »Uncharted 4«-Ankündigung im Köcher, ob der Pfeil sein Ziel findet, werden wir aber ebenfalls erst 2015 erfahren. Bis dahin werden es die Multiplattform-Titel richten müssen, deren Präsentationen Sony und Microsoft fast schon brüderlich unter sich aufteilten. Metal Gear Solid 5 gab's bei Sony zu sehen, das neue Call of Duty dafür bei Microsoft. Ubisoft zeigte Far Cry 4 auf der Playstation 4, The Division im Gegenzug auf der Xbox One. Erneut hofierten beide Konsolenhersteller die Indie

Developer, erneut blieben beide konkrete Details zu ihren jeweiligen Self Publishing Modellen schuldig. Ebenso mutig wie überraschend war lediglich Sonys eindeutiges Bekenntnis zu Free2Play-Spielen.

Fast scheint es, als hinge die NextGen immer noch in den Startlöchern fest. Offenkundig haben viele Publisher den neuen Konsolen nicht viel zu getraut und – überrascht vom Verkaufserfolg – erst letztes Jahr ihre AAA-Produktionsmaschine in Gang geworfen. Besonders offensichtlich war das bei EA, die fast schon verlegen lediglich Prototypen von Star Wars Battlefront, Mirror's Edge 2 oder dem neuen Criterion-Spiel zeigten. Erstaunlicherweise könnte also das von vielen schon abgeschriebene Nintendo zum Gewinner im E3-Konsolenkampf 2014 werden. Denn hier hat's mit Mario Kart 8 schon vor der Messe genau den System Seller gegeben, auf den Sony und Microsoft immer noch warten. Und zwar im Handel!

Auch für deutsche Games-Entwickler sind die NextGen-Startschwierigkeiten prinzipiell eine gute Nachricht. Denn je mehr Lücken die großen Konsolenhersteller und Publisher hinterlassen, desto mehr Chancen bieten sich kleineren, agileren Teams. Yager hat's mit der überraschenden Übernahme der Dead-Island-Franchise schon mal vorgemacht. Mehr dazu demnächst in diesem Magazin. **Heiko Klinge**

WWW.makinggames.de
■ aktuelle News und Gerüchte



RESPAWN KOOPERIERT MIT GDC EUROPE

In einer Pressemitteilung wurde die Kooperation der Veranstaltung »Respawn – Gathering of Game Developers« mit der GDC Europe bekannt gegeben. Besucher der Respawn erhalten mit ihren Tickets (ab 79 Euro) gleichzeitig Zutritt zum Ausstellungsbereich der GDC Europe sowie einen Rabatt von 10 Prozent auf den GDC Europe All Access Pass, der in der Early-Bird-Phase bis 16. Juli noch 827 Euro kostet und anschließend mit 946 Euro zu Buche schlägt (Preisangaben inklusive Mehrwertsteuer). Im Gegensatz zur GDC in San

Francisco gibt es bei der GDC Europe keinen separaten Expo Pass, daher ist ein Ticket für die Respawn für all diejenigen die günstige Variante, die lediglich am Ausstellungsbereich und am Networking interessiert sind. Außerdem richten die beiden Veranstaltungen am 12. August erstmals gemeinsam die »GDC Europe and Respawn Developer Night« aus. Die Respawn findet zum zweiten Mal statt und bietet Teilnehmern am 12. und 13. August im DOCK.ONE in Köln Vorträge, Workshops und Diskussionen.

6 ONLINE-STUDIOS WECHSELN ZUM BIU

Ende Mai hat der BIU – Bundesverband Interaktive Unterhaltungssoftware bekanntgegeben, dass er sechs neue Mitglieder aufgenommen hat. Dabei handelt es sich um einige der größten deutschen Entwickler für Browser- und Mobilespiele: Gameforge, Goodgame Studios, InnoGames, SevenGames, upjers sowie Wooga.

Bei der ordentlichen Mitgliederversammlung des BIU Anfang Juni wurde nun der Vorstand des Verbands neu gewählt. Dr. Klemens Kundratitz (Koch Media) wurde dabei im Amt bestätigt. Neu in den Vorstand gewählt wurden Dr. Bernd Fakesch (Nintendo of Europe) sowie mit Tobias Haar (Gameforge) auch ein Vertreter der Neuzugänge. »Dass mit Gameforge eines der neuen BIU-Mitglieder direkt in den Vorstand gewählt wurde, zeigt, wie ernst dem BIU eine Weiterentwicklung der Verbandsarbeit in Richtung der Publisher und Entwickler von Online- und Mobile-Spielen ist«, sagt Haar über die Vorstandswahlen.



Vier Fragen an ...

Tobias Haar
ist Mitglied der
Geschäftsleitung
von Gameforge.



Making Games Wieso hat sich Gameforge für einen Beitritt zum BIU und gegen den G.A.M.E.-Verband entschieden?

Tobias Haar Wir haben uns für einen Eintritt in den BIU entschieden, da wir in der Zusammenarbeit mit dem Verband die besten Chancen sehen, die Themen der gesamten Spielebranche in den nächsten Jahren aktiv voranzutreiben. Da die angedachte Fusion zwischen G.A.M.E und BIU kürzlich leider gescheitert ist, mussten wir uns für einen Verband entscheiden. Dabei kam unseren Anforderungen die Ausrichtung und Aufstellung des BIU am nächsten.

Making Games Welche Themen wollt ihr in den nächsten Monaten gemeinsam mit dem BIU angehen?

Tobias Haar Für uns ist es wichtig, dass die Akzeptanz von digitalen Spielen in Politik und Gesellschaft gesteigert wird, um den Spiele-Standort Deutschland noch besser zu etablieren und zum Vorteil aller weiter voranzubringen. Das umfasst nach unserer Überzeugung auch, dass man neuen Geschäftsmodellen gegenüber offen sein muss, beziehungsweise dass wir als Vertreter der Branche aktiver in die Diskussion mit Politik, Wirtschaft und Gesellschaft gehen. Hier ist in den letzten Jahren gerade durch Verbandsarbeit schon sehr viel erreicht worden. Das gilt es nun

gemeinsam fortzusetzen – Zeit zum Ausruhen gibt es nicht.

Making Games Wie wird in Zukunft eure Zusammenarbeit mit dem G.A.M.E. aussehen?

Tobias Haar Unsere Ziele wie die Stärkung des Spiele-Standorts Deutschland und die weitere Steigerung der Akzeptanz von digitalen Spielen in Politik und Gesellschaft teilen sicherlich auch weiterhin viele Mitglieder des GAME. Daher ist es sinnvoll, auch künftig mit gemeinsamen Projekten zum Beispiel auf Verbandsebene an diesen Zielen zu arbeiten. Ein gutes Beispiel hierfür ist sicherlich der Deutsche Computerspielpreis, der zu einer wichtigen Veranstaltung geworden ist, wenn es um die Sichtbarkeit unserer Branche über die Spieler-Community in Deutschland hinausgeht.

Making Games Welche Auswirkungen hat der BIU-Beitritt auf die Organisation der gamescom?

Tobias Haar Für die gamescom 2014 sind wir natürlich mit unseren Planungen schon sehr weit fortgeschritten. Allerdings bietet der BIU natürlich für Mitglieder zahlreiche Vorteile, die wir in Zukunft vermehrt in Anspruch nehmen werden, wie zum Beispiel die Mitarbeit an der Gestaltung der gamescom über die BIU Arbeitskreise sowie zusätzliche Serviceleistungen rund um die gamescom.

Eventkalender

JULI 2014

Develop Conference

Ort Brighton, Hilton Brighton Metropole
Termin 08. bis 10. Juli 2014
Thema Konferenz
Kosten Eintritt frei bis 918 Euro
Homepage www.developconference.com

Making Games Talents

Ort Frankfurt, IHK Frankfurt
Termin 19. Juli 2014
Thema Recruiting Event
Kosten 29 Euro
Homepage www.makinggames.de/talents

Casual Connect USA

Ort San Francisco, Hilton San Francisco Union Square
Termin 22. bis 24. Juli 2014
Thema Konferenz
Kosten 367 bis 1837 Euro
Homepage usa.casualconnect.org

AUGUST 2014

GDC Europe

Ort Köln, Koelnmesse
Termin 11. bis 13. August 2014
Thema Konferenz
Kosten 177 bis 1065 Euro
Homepage www.gdceurope.com

Respawn - gathering of game developers

Ort Köln, DOCK.ONE
Termin 12. bis 13. August 2014
Thema Konferenz
Kosten 79 bis 189 Euro
Homepage www.respawngathering.com

gamescom

Ort Köln, Koelnmesse
Termin 13. bis 17. August 2014
Thema Messe
Kosten 11,50 bis 28 Euro
Homepage www.gamescom.de

Unite 2014

Ort Seattle, Marion Oliver McCaw Hall
Termin 20. bis 22. August 2014
Thema Konferenz
Kosten 79 bis 264 Euro
Homepage unity3d.com/unite/unite2014

SEPTEMBER 2014

IFA

Ort Berlin, Berlin Expo Center City
Termin 05. bis 10. September 2014
Thema Messe
Kosten 8 bis 35 Euro
Homepage b2c.ifa-berlin.de

WWW.makinggames.de

Online-Eventkalender mit Suchfunktion und Veranstaltungsinfos

Passiert auf makinggames.de

Bald ein eigenes Early-Access-Programm von GOG?



Auch wenn es bei Steam zuletzt einige Rückschläge in Form potentieller Betrugsmaschinen wie etwa bei »Earth: Year 2066« gab, denkt man mittlerweile bei anderen Anbietern darüber nach, das Early-Access-Konzept aufzugreifen – so etwa bei GOG.com. Oder zumindest deutete Marcin Iwinski, Mitbegründer des polnischen GOG-Betreibers CD Projekt, dies an. Man habe das Phänomen natürlich im Blick und denke darüber nach, es auch auf GOG.com anzuwenden – dann allerdings in stark abgewandelter Form. Immerhin sei auch das Grundkonzept der eigenen Plattform ein anderes als das von Steam. Man selbst setze nur auf Titel ohne DRM und verwalte das Angebot an Spielen zudem gewissenhaft. Er selbst, so der CD-Projekt-Manager, fühle sich in den ausufernden Angeboten anderer Plattformen häufig völlig verloren. Er glaube deshalb, so Iwinski weiter, dass es sinnvoller sei, die Qualität und Funktionalität der angebotenen Spiele vor einer Veröffentlichung auf GOG.com zu kontrollieren. Schließlich fühle er sich verantwortlich dafür, was die Leute auf seiner Plattform kauften. Außerdem möchte er seine Kunden vor Fehlkäufen und falschen Versprechungen durch die Entwickler schützen. Deshalb halte er es für sinnvoll, den Spielern beispielsweise die Möglichkeit zu geben, den Kauf einer Alpha-Version wieder rückgängig zu machen, sollte die Weiterentwicklung des Spiels nicht so verlaufen, wie vom Entwickler versprochen.



Samsung arbeitet an eigener VR-Brille

Medienberichten zufolge arbeitet Samsung an einer eigenen VR-Brille. Diese ist laut den Meldungen sowohl billiger als auch besser als die bereits bekannten Projekte anderer Anbieter, namentlich Oculus Rift von Oculus VR und Project Morpheus von Sony. Erste Entwickler sollen bereits Dev-Kits von Samsung erhalten haben. Die VR-Brille werde jedoch kein eigenes Display haben, sondern eine Halterung für Smartphones sein. Dabei kooperiert Samsung offenbar eng mit Oculus VR: Das Unternehmen soll vor allem sein SDK für Mobilgeräte einbringen. Im Gegenzug bekomme Oculus von Samsung Zugriff auf hochauflösende OLED-Displays – Auflösungen von mehr als Full-HD wie bei der DK2-Entwicklerversion sind im Gespräch.

Microsoft hat die App-Richtlinien verschärft



Microsoft

Wie Microsoft im offiziellen Unternehmensblog bekannt gegeben hat, will man in Zukunft verstärkt gegen Entwickler von Apps vorgehen, die sich bei der Namens- oder Logofindung für ihre Produkte zu stark an bereits existierenden Apps orientieren. Unter diese neuen Richtlinien fallen auch Apps, die als Anleitungen für Spiele gedacht sind. Einen »Angry Birds Star Wars 2 Game Guide« dürfte es beispielsweise in Zukunft nicht mehr geben; die App müsste stattdessen »Game Guide for Angry Birds Star Wars 2« heißen. Mit den neuen Richtlinien will Microsoft Verwechslungen vorbeugen und unabsichtliche Downloads verhindern.

Zenimax reicht Klage gegen Oculus VR ein

In der Klageschrift wird Oculus VR und Gründer Palmer Luckey vorgeworfen, »illegal und widerrechtlich Geschäftsgeheimnisse von Zenimax, die sich auf Virtual Reality beziehen, zu nutzen und Rechte sowie Marken von Zenimax zu verletzen«. Zenimax erklärt außerdem, dass das Virtual-Reality-Headset »Geschäftsgeheimnisse, urheberrechtlich geschützten Computer-Code und technisches Wissen« nutzt, die Zenimax gehören und über die eine Schweigevereinbarung mit Oculus VR abgeschlossen worden seien. Oculus habe jedoch alle Entschädigungsforderungen abgelehnt. Palmer Luckey habe sich zwar aus der Öffentlichkeit herausgehalten und sich als »visionärer Entwickler der VR-Technik« dargestellt, doch tatsächlich sei die grundlegende Technik zur Herstellung von Oculus Rift von Zenimax entwickelt worden. Oculus VR hatte schon vor der Klage erklärt, dass die Vorwürfe jeglicher Realität entbehren. Auch John Carmack, der von der Zenimax-Tochter id Software zu Oculus VR wechselte, hatte dies laut eigenen Worten getan, weil er bei Zenimax nicht an Virtual Reality arbeiten durfte. Kritiker merkten auch an, dass Zenimax sich erstaunlicherweise erst dann meldete, nachdem Facebook die milliardenschwere Übernahme von Oculus VR angekündigt hatte, obwohl Oculus Rift schon sehr viel länger in der Öffentlichkeit steht.



Google will Twitch kaufen



Rund eine Milliarde Dollar will Google sich die Übernahme

der Streaming-Plattform Twitch kosten lassen. Bei einem erfolgreichen Deal soll Twitch diversen Medienberichten zufolge in das Video-Portal YouTube integriert werden. Eine offizielle Bestätigung gibt es bisher weder von Google noch von Twitch, doch gerüchteweise soll die Übernahme schon bald offiziell bekannt gegeben werden. Twitch wurde 2011 von Justin Kan und Emmett Shear gegründet und verzeichnete vor allem im vergangenen Jahr ein starkes Wachstum. Monatlich verzeichnet die Streaming-Plattform nach eigenen Angaben inzwischen mehr als 45 Millionen Besucher, über eine Million User seien aktiv am Upload von Videos beteiligt.



EA schließt Mythic Entertainment

EA hat sein MMO- und Rollenspiel-Entwicklerstudio Mythic Entertainment geschlossen. In einem ersten Statement wird die Schließung so begründet: »Wir schließen den Standort von EA Mythic in Fairfax, Virginia, weil wir unsere Mobile-Entwicklungen auf unsere anderen Studio-Standorte konzentrieren. Wir arbeiten mit allen betroffenen Angestellten zusammen, um sie im Rahmen einer kommenden Job-Messe bei der Suche nach neuen Job-Möglichkeiten zu unterstützen – sei es innerhalb von Electronic Arts oder bei anderen Unternehmen.« Es ist nicht bekannt, wie viele Mitarbeiter von der Mythic-Schließung betroffen sind. Das Studio wurde 1995 als Interworld Productions gegründet und 1997 umbenannt. Der Durchbruch gelang dem Unternehmen erst 2001 mit dem Online-Rollenspiel »Dark Age of Camelot«. 2006 erfolgte die Übernahme durch EA sowie ein zwischenzeitlicher Zusammenschluss mit BioWare zu BioWare Mythic, der 2012 wieder rückgängig gemacht wurde.

Gamigo wird für Wechsel des Börsensegments kritisiert



Am 20. Juni 2014 wird Gamigo das Börsensegment vom Entry Standard in ein Listing im Quotation Board wechseln und muss dafür Kritik einstecken. Als Grund für den Wechsel gibt das Unternehmen die höheren Kosten im Entry-Standard-Segment an. Dieses Argument will unter anderem Michael Hedtstück vom Finance Magazin nicht gelten lassen. Er bezeichnet diesen Wechsel als »dreiste Abzocke seit langem.« Das Problem daran ist für Hedtstück nicht nur der immense Wertverlust der Aktie in Höhe von 50 Prozent in den vergangenen Monaten. Er kritisiert in seinem Kommentar vor allem die Begründung des Wechsels und die Tatsache, dass sich Gamigo dadurch in ein Börsensegment begibt, in dem es vor dubiosen Firmen nur so wimmelt. Hinzu komme außerdem noch, dass das Rating von Gamigo ausgelaufen sei und auch nicht mehr erneuert werde.



Philips zieht gegen Nintendo vor Gericht

Nintendo hat in seinen Wii-Konsolen angeblich Technologien verwendet, auf die der Elektronikkonzern Philips die Patente hält. Das behauptet zumindest Philips und zog nun vor Gericht. Konkret soll es sich um ein Patent für ein »Gerät zur virtuellen Körper-Steuerung« handeln, mit dem sich Steuerungs-Schnittstellen durch intuitive Bewegungen bedienen lassen. Eine Beschreibung, die allerdings auch auf den Kinect-Sensor von Microsoft zutreffen würde – der Konzern aus Redmond ist jedoch nicht Gegenstand der Klage. Darüber hinaus soll Nintendo mit seiner Wii Remote ein zweites Patent verletzen, das sich auf ein »User-Interface-System basierend auf einem Zeige-Gerät« bezieht. Laut Philips wurde Nintendo bereits im November 2011 über die Verletzung des ersten Patents informiert. Die zweite Patentverletzung hingegen wurde bisher wohl noch nicht öffentlich gemacht. Der Elektronikkonzern verlangt vom japanischen Konsolenhersteller nun Schadensersatz und ein US-Verkaufsverbot für sämtliche Nintendo-Produkte, die die beiden Patente verletzen.



Kinect-Kamera + Wii-Balanceboard + Oculus Rift = Hoverboard VR.
www.makinggames.de/hoverboard-vr

Raph Koster's Gedanken zur finanziellen Zukunft von kleinen Entwicklerstudios.
www.makinggames.de/financial-future

Im Trailer zum Spiel »Glorious Leader« reitet Kim Jong Un auf einem Einhorn und wird von Dennis Rodman unterstützt.
www.makinggames.de/glorious-leader

Ein Rant von Jeremiah Mueller zum Thema »Videospiele können gewalttätig machen«.
www.makinggames.de/video-game-violence



BRANCHEN-GEZWITSCHER @MAKINGGAMES_DE/INTERESTING-PEOPLE



@GEORGE3DR

Holy shit...getting on Steam really is sort of like winning The Hunger Games.
It's so clear now. :)

George Broussard, Erfinder von Duke Nukem



@WILL_LUTON

Current indie generation slowly transitioning to become the new old guard. Ideals forgotten.

Will Luton, Autor von »Free-to-Play«



@THA_RAMI

It's so discouraging to have a sexism discussion on your Facebook & realise that so many people still don't understand there's a problem.

Rami Ismail, Mitgründer von Vlambeer



@THEREALCLIFFYB

Best part about Epic's plans for UT is that if you have a problem with it you can shut the fuck up and fix it yourself. :)

Cliff Bleszinski, Lead Designer von Unreal Tournament



@GRUMPYGAMER

I really really really really really really really really don't want to learn DirectX.

Ron Gilbert, Erfinder von Monkey Island



@NOTCH

Writing gui code is so much fun. Also I like to hug and kiss poisonous snakes.

Markus Persson, Minecraft-Erfinder



@BENJAMINCOUSINS

Bethesda goes F2P. The last of the mighty beasts has been slain.

Ben Cousins, Independent Videogames Consultant



@CHRIS_AVELLONE

South Park Stick of Truth employee reviews are comedy gold when you have to refer to someone's fine work in Mr. Slave's Ass.

Chris Avellone, Obsidian Entertainment



@LARALYN

Personally, I love deep story and meaningful content in games... but I don't want to feel guilty for enjoying the primary mechanic of a game.

Laralyn McWilliams, Creative Director bei The Workshop

JOIN THE CONVERSATION

TOOLS

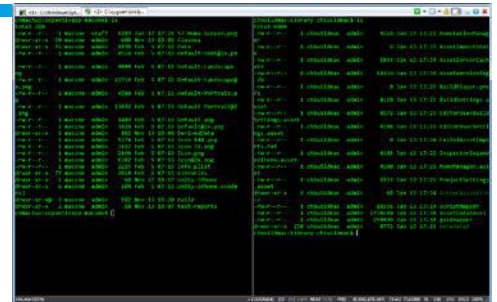
Keine guten Werkzeuge – keine guten Spiele! Experten aus der Branche sagen ihre Meinung zu aktuellen Programm-Updates sowie neuen Tools und verraten ihre persönlichen Software-Geheimtipps.

CONEMU [CODE.GOOGLE.COM/P/CONEMU-MAXIMUS5](http://code.google.com/p/conemu-maximus5)



Andreas Katzig
ist Technical Director
bei Chimera Entertainment.

Wer Spiele programmiert, nutzt auch Command Line Tools. Gerade wer auf und für verschiedene Plattformen entwickelt, darf sich mit der Shell auf Mac OS X, Linux oder Windows auseinandersetzen. Kommandozeilenhacker wissen das Mac-OSX-Terminal zu schätzen, das mit seinen Tabs die Produktivität steigert. Was es schon lange für den Mac gibt, gehört leider nicht zum Standardumfang von Windows. Hier hilft ConEmu: Der kostenlose Emulator bietet eine erweiterte Oberfläche für Shells. ConEmu erlaubt nun, durch Tabs separiert mehrere dieser Shells in seinem Hauptfenster anzuzeigen. In Kombination mit Cygwin, dessen Bash man am besten mit dem Befehl »C:\cygwin\bin\bash.exe -login« startet, sorgt ConEmu unter Windows für echtes Linux- oder Mac-Feeling.



Der Emulator ConEmu bietet durch seine Tab-Aufteilung im Hauptfenster eine bequeme Darstellung von Shells.



Sowohl im 2D- als auch 3D-Bereich ist Lazy Nezumi Pro ideal für geschwungene Linien oder Ornamente.

LAZY NEZUMI PRO [WWW.LAZYNEZUMI.COM](http://www.lazynezumi.com)

Lazy Nezumi Pro (LN) glättet Pinselstriche in Photoshop, wobei neben den Werten der Pinselposition auch die Druckempfindlichkeit geglättet werden kann. Das Tool integriert sich in Photoshop als Plugin, kann aber für jede andere Applikation manuell aktiviert werden und leistet so auch beim Sculpting (Zbrush) gute Dienste. LN ist sowohl im 2D-, als auch 3D-Bereich ideal für geschwungene Linien, Line-Art oder Ornamente. Die Pinselstrichglättung ist stufenlos anpassbar, kann per Shortcut aktiviert / deaktiviert werden und bietet die Möglichkeit, eigene Tool-Einstellungen zu speichern. Ein weiteres Plus ist der sehr günstige Preis von 29,99 \$ für eine kommerzielle Lizenz.



Alexander Asmus
ist Senior Artist
bei Funatics Software.

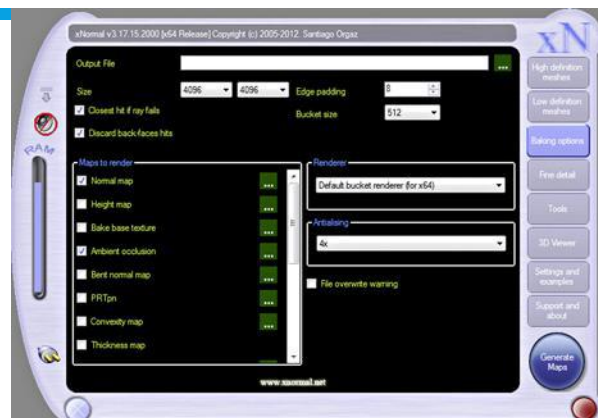
XNORMAL [WWW.XNORMAL.NET](http://www.xnormal.net)



Bleick Bleicken
ist Grafik-Freelancer
bei Pixelstorm.

Man darf sich nicht täuschen lassen:

Das 3D-Tool xNormal kommt im sehr fragwürdigen Retro-Design daher – aber unter der Haube verbirgt sich eine ansehnliche Sammlung nützlicher Funktionen, die das Rendern von Maps erleichtern. Überwiegend wird xNormal für die Berechnung von Normal- und Ambient Occlusion Maps verwendet. Zusammen mit einem importierten Cage und der Option, die GPU rendern zu lassen, bekommt man in erstaunlich kurzer Zeit fehlerfreie Maps. Zusätzlich bietet xNormal Photoshop Filter einen 3D Viewer und zahlreiche weitere, teils exotische Map-Formate, wie »Thickness« zur Simulation von Subsurface Scattering Effekten. Und das Beste dabei: es ist kostenlos.



xNormal erleichtert das Rendern und die Berechnung von Normal- und Ambient Occlusion Maps.

FOKUS G.A.M.E.

»DIGITALE AGENDA: DEUTSCHER COMPUTERSPIELPREIS«

Quo vadis, DCP? G.A.M.E.-Geschäftsführer Thorsten Unger über die geänderten Zuständigkeiten beim Deutschen Computerspielpreis und darüber, an welchen Schrauben gedreht werden muss, damit der deutsche Spielemarkt im weltweiten wirtschaftlichen Vergleich eine Rolle spielen kann.

Nach dem Computerspielpreis ist vor dem Computerspielpreis. Zum sechsten Mal wurde am 15. Mai 2014 die in Branchenkreisen »DCP« bezeichnete Auszeichnung vergeben. Unter Teilnahme von maßgeblicher politischer Prominenz, jedoch ohne Glamour und mit dem ein oder anderen Skandalchen im Vorfeld ging die Veranstaltung recht unspektakulär und emotionslos über die Bühne. »Same procedure as every year«, mag man meinen. Wenn nicht so manches Detail Hoffnung auf einen Umbruch und einen Neuanfang machen würde.

Geänderte Zuständigkeit

Staatssekretärin Bär, eine Gamerin, hat schon 2007 – übrigens unter anderen mit den beiden Monikas Griefahn und Grütters – den Antrag zur Vergabe eines Computerspielpreises in den Bundestag eingebracht. Während Frau Griefahn heute dem Beirat der Stiftung Digitale Spielkultur vorsitzt, hat Frau Grütters nach Amtsantritt die Zuständigkeit für eben unseren »DCP« an das neu geschaffene Ministerium für Verkehr und – oh, wo sind Kultur und Medien geblieben? – Digitale Infrastruktur übergeben.

Da gibt es zum einen den Bundesminister Alexander Dobrindt, zum anderen Frau Bär. Beide sind nach unserem Branchenverständnis Kernzielgruppe. Das ist erst einmal für das Branchenverständnis gut, denn es macht schon Sinn, jemanden innerhalb der Bundesregierung mit einem Thema zu betrauen, der sich auch angesprochen fühlt. Das war bei Herrn Bernd Neumann nicht unbedingt der Fall. Seinem Engagement für unsere Sache sei ihm auch wegen mancher Kontroverse sehr gedankt. Er ist aufgrund seiner Überzeugung folgerichtig nun Präsident der staatlichen Filmförderanstalt (FFA) und weiter bemüht, die Filmwirtschaft mit Förderung zu versorgen. So hat sich das Alter unserer Ansprechpartner deutlich verjüngt. Von personeller Seite also schon mal eine gute Basis.

Aber was haben Games mit dem Ministerium für Digitale Infrastruktur zu tun? Mehr als man denkt, wenn man sich einmal mit dem Thema

auseinandersetzt, denn irgendwann dämmerte es auch in Berlin: Wieso heißen die Marktführer im weltweiten IT-Geschäft nicht 1&1, Freenet oder Zalando, sondern Google, Microsoft oder Amazon? Und, Moment mal, wieso heißen die globalen Marktführer im Spielesektor nicht Gameforge, Deck 13 oder rondomedia, sondern Activision Blizzard, Ubisoft und Electronic Arts? Zum Thema »Digitale Agenda« schreibt die Bundesregierung Folgendes: »Wir wollen Deutschland zum führenden Standort der digitalen Wirtschaft in Europa machen.« Das eben ist die Hauptaufgabe von Bundesminister Alexander Dobrindt, neben der Instandsetzung und Fitmachung unserer Straßen und Wasserwege.

Der Exportweltmeister schwächelt

Klare Ansage. So sieht die aktuelle Lage aus: Deutschland stellt laut Marktforschungsinstitut Newzoo heute 5,5% des Weltmarktes an den Konsumentenausgaben für Games bei etwa 81,5 Milliarden Dollar. Aber deutsche Unternehmen setzen nicht mal eine Milliarde davon selbst um. Der Exportweltmeister und Primus vieler Branchen hat wahrscheinlich nicht einmal 1% am Weltmarkt, und das in einem Hochtechnologie-sektor mit bekannten Abstrahlungseffekten in die klassische Wirtschaft (Simulationstechnologie, Mensch-Maschine-Interface etc.).

Das ist also die Aufgabe, der wir uns stellen müssen: Wie ändern wir dieses Verhältnis? Welche Bedingungen müssen wir schaffen, damit das nächste »Grand Theft Auto« wirklich mal aus Deutschland kommt? Zum Beispiel durch einen zeitgemäßen Computerspielpreis. Einen »DPC«, der weder Alterskennzeichnungen, Kategorie-Einschränkungen noch längst veröffentlichte Spiele zum Gegenstand hat. Der Deutsche Computerspielpreis muss die Bühne deutscher Produktionen sein, eine Leistungsschau der deutschen Branche. Ohne Wenn und Aber. Und natürlich darf die Kultur nicht fehlen. Die steht auch in der digitalen Agenda, eben in dem für uns relevanten Handlungsfeld »Digitale Gesellschaft, Forschung, Bildung und ...« – genau – »Kultur«. Dann kann ja nicht mehr viel schiefgehen, oder?

Thorsten Unger



Thorsten Unger
ist Geschäftsführer beim
G.A.M.E. Bundesverband der
Computerspielindustrie e.V.

Thorsten Unger ist Ansprechpartner in allen politischen Fragen und setzt sich für die Belange des Verbands in Bezug auf die wirtschaftliche, gesellschaftspolitische und kulturelle Bedeutung von Games ein.



Der DCP muss zur Bühne für aller deutschen Produktionen werden – ohne Kategorie-Einschränkungen oder Alterskennzeichnungen.



Making Games 04/2014



DAS ÄNDERT ALLES

Innovation wird überschätzt. Und Innovation ist unverzichtbar. Beides stimmt – je nachdem, wie man Innovation versteht. Die Vorstellung von der Idee, die alles verändert, ist ein großes Missverständnis.



Christian Schmidt
ist Lead Market Research
Analyst bei Bigpoint.

Christian leitet bei Bigpoint die Kundenforschung und Marktanalyse, zu der auch die Beobachtung von Trends und Innovationen gehört. Bis 2011 war er stellvertretender Chefredakteur von GameStar. Gemeinsam mit Gunnar Lott würdigt er im Retro-Podcast »Stay Forever« jeden Monat Spieleklassiker.
www.stayforever.de
www.youtube.com/user/StayForeverDE

Innovativste Spiele?

Die bisherigen Preisträger des D.I.C.E.-Awards »für herausragende Innovation« seit der Einführung der Kategorie im Jahr 2005, vergeben von der amerikanischen Academy of Interactive Arts and Sciences:

- 2005:** Nintendogs
- 2006:** Wii Sports
- 2007:** Rock Band
- 2008:** LittleBigPlanet
- 2009:** Scribblenauts
- 2010:** Heavy Rain
- 2011:** Skylanders: Spyro's Adventure
- 2012:** Journey
- 2013:** The Last of Us

Die Oscar-Verleihung der Spieleindustrie heißt D.I.C.E. Awards. Nicht ganz so glamourös wie ihr Vorbild, versammelt sie nichtsdestotrotz die Creme de la Creme der internationalen Spieleindustrie – so auch Anfang Februar 2014, als im Hard Rock Hotel & Casino in Las Vegas Felicia Day und Freddie Wong vor rund 600 Gästen auf die Bühne treten und die Vergabe von 24 Trophäen einläuten. Eine davon ist der Preis »für herausragende Innovation«.

Die Auszeichnung wird von der Academy of Interactive Arts and Sciences, die die D.I.C.E. Awards ausrichtet, seit 2005 vergeben. Sie belohnt, in den Worten der Academy, »das Spiel, das durch Innovation in Design und Ausführung am wahrscheinlichsten eine neue Kategorie von Spielen definiert, oder einen kreativen oder technologischen Durchbruch in einem existierenden Genre darstellt.« Auf der Bühne in Las Vegas hält der Oculus-Gründer Palmer Luckey gemeinsam mit Unity-Chef David Helgason die Laudatio. Er kann sich das Lachen nicht verkneifen, als er mit ironischem Pathos ins Publikum ruft: »Es gibt Menschen, die Spiele machen, und Menschen, die Spiele für immer verändern!« Helgason nestelt den Umschlag auf, er verliert den Sieger: »The Last of Us!« Jubel im Saal.

Moment, »The Last of Us«? Das innovativste Spiel des Jahres 2013? Während gleichzeitig »Papers, Please«, »Tearaway« und »The Stanley Parable« nominiert waren?

Luckeys Ironie ist angemessen. Kaum jemand dürfte glauben, dass »The Last of Us« Spiele

für immer verändert. Und man kann daran zweifeln, ob es einen kreativen Durchbruch darstellt, geschweige denn eine neue Kategorie von Spielen definiert. Die diesjährige Wahl der Academy wird der hochtrabenden Ambition der Preiskategorie nicht gerecht. Und das ist gut so.

Anspruch und Wirklichkeit

Denn wenn es um Innovation geht, demonstriert kaum etwas besser den Unterschied zwischen Anspruch und Wirklichkeit als die Liste der bisherigen D.I.C.E.-Preisträger (siehe Kasten). Es ist eine vernünftige Auswahl von Titeln, deren innovatives Element sofort einleuchtet; von »Nintendogs« bis »Journey«, von »Scribblenauts« bis »Skylanders« hat jedes Spiel die Auszeichnung verdient. Keines jedoch strahlt – vielleicht mit der Ausnahme von »Skylanders« – Wirkungskraft bis in die Gegenwart aus. Das Musikspiel-Genre rund um »Rock Band« ist Schnee von gestern, der Bewegungshype um »Wii Sports« abgeklungen, »Scribblenauts« steht als kreatives Kuriosum für sich allein. Die innovative Kraft all dieser Spiele reichte dafür, sich mehr oder weniger große Marktnischen zu schaffen, die sie weitgehend selbst ausfüllen. Zu echter Innovation fehlt ihnen die Nachhaltigkeit. So steht die D.I.C.E.-Liste exemplarisch für den Widerspruch zwischen dem, was als Innovation empfunden wird, und dem, was Innovation ist.

Was ist Innovation?

In den meisten Köpfen wird die Vorstellung von Innovation bestimmt vom revolutionär Neuen; dem Nie-Dagewesenen, das die Grenzen

des Machbaren erweitert. Solche Ereignisse existieren zweifellos. Aber weder sind sie die einzige Form von Innovation, noch die wichtigste. Wohl aber diejenige, die am häufigsten bedeutungslos bleibt. »Eine Innovation, die sehr groß aussieht, mag sich als nichts anderes als technische Virtuosität herausstellen«, schreibt der Ökonom Peter Drucker im Buch »Innovation and Entrepreneurship«, »und Innovationen mit bescheidenen intellektuellen Ambitionen können zu gigantischen, hochprofitablen Unternehmungen heranwachsen, so wie zum Beispiel McDonald's.« Die Spielegeschichte ist voll von gescheiterten Revolutionen. Peter Molyneux's »Black & White«, das autonome künstliche Intelligenz erschaffen wollte; Activations »Battlezone«, das Echtzeit-Strategie und Ego-Shooter zu verschmelzen versprach; Microsofts »Kinect«, das den Controller eliminiert; »Scribblenauts«, das Worte in Spielobjekte übersetzt: technische Virtuosität. »Innovation ist überbewertet«, sagte der »Diablo 3«-Designer Jay Wilbur 2008 berühmterweise im GameStar-Interview. Was er meinte war: Diese Art von Innovation – die konventionsbrechende Innovation – ist überbewertet. Das Machbare muss noch lange nicht das Sinnvolle sein. Es fasziniert uns ungemein, wenn eine Innovation eine technische Hürde überspringt, die als unüberwindlich galt, oder wenn sie uns eine Erfahrung verspricht, die uns reizvoll neu erscheint. Aber dieser Schauwert, der unsere Neugierde bedient, verpufft in Windeseile, wenn sich ihm kein Nutzwert zugesellt.

Es gibt deshalb auch einen anderen, nüchternen Blick auf Innovation; nämlich einen, der nicht das erfolglose Revolutionäre betrachtet, sondern das erfolgreiche Gewöhnliche. Auf dessen Liste nicht »Rock Band«, »Scribblenauts« oder »Heavy Rain« stehen, sondern »Clash of Clans«, »World of Warcraft« oder »Candy Crush Saga«. Der fragt, welche Innovation in diesen Titeln steckt, die es ihnen erlaubt, ihre Konkurrenz auszusteichen.

»Clash of Clans«, »World of Warcraft«, »Candy Crush Saga« – keines dieser Spiele hat sein Segment um bahnbrechende Neuerungen bereichert. Ihr Schöpfungsbeitrag war stattdessen, bekannte Rezepte auf eine Weise aufzubereiten,



»The Last of Us« wurde 2013 bei den D.I.C.E.-Awards für »herausragende Innovation« ausgezeichnet. Zwar erfindet das Spiel nichts neu, setzt jedoch mit seiner ambitionierten Erzählweise neue Maßstäbe.

die sie dem Massenmarkt erschließen. Dadurch entfalten sie maximale Wirkungskraft.

Zugänglichkeit als Innovation

Zugänglichkeit ist keine glamouröse Art von Innovation. Im Gegenteil: Sie widerspricht der konventionellen Vorstellung von Innovation, die eine Innovation des Hinzufügens ist. Zugänglichkeit bedeutet meist eine Innovation des Weglassens. Von der avantgardistischen Freiheit eines »Ultima Online«, seinen Spielcharakter beliebig zu entwickeln und in einer gleichsam anarchischen Spielwelt jeden Spieler bekriegen zu können, ist in »World of Warcraft« nicht mehr viel übrig. Das hochkomplexe Rollenspiel-System eines »Everquest« und dessen harte Konsequenz beim virtuellen Tod hat »World of Warcraft« abgemildert. Und damit das Genre der Online-Rollenspiele einer breiten Spielerschaft geöffnet. Eben jener Ausleseprozess – Bestehendes dadurch weiterzuentwickeln, indem man Teile davon aufgibt – besitzt innovative Kraft. Aber das ist nicht alles.

Optimierung des Bestehenden

Dazu kommt eine weitere Absetzung vom traditionellen Verständnis von Innovation, nämlich der Innovation als Streben nach dem Bestmöglichen. »World of Warcraft« oder »Candy Crush Saga« sind keine bestmöglichen Spiele. Sie reizen beileibe nicht die Grenzen des Machba-



Der Innovationsgehalt von Spielen wie »Rock Band« (links) und »Scribblenauts« (rechts) reichte zur Erschließung von Marktnischen, war jedoch nicht nachhaltig.

ren aus, schon gar nicht technisch. »World of Warcraft« wurde schon bei seinem Erscheinen im Jahr 2004 als visuell rückständig kritisiert. Zehn Jahre und Hunderte von gleich aussehenden Online-Spielen später, muss das Urteil lauten: Der stilisierte Look war pure Avantgarde. In der visuellen Gestaltung von »World of Warcraft« drückt sich die Erkenntnis aus, dass Menschen die grundlegende Tendenz haben, Abstriche bei der Qualität in Kauf zu nehmen, wenn sie dadurch Komfort gewinnen. Live-Musik ist hochwertiger als CD-Musik, CD-Musik hochwertiger als Online-Streaming, und doch wiegt die Allverfügbarkeit von mp3 & Co. deren Qualitätsnachteil mehr als auf. In gleichem Maß bedient die Optik von »World of Warcraft« das Bedürfnis, das Spiel leichter lesen, besser verstehen und flüssiger spielen zu können. Eine Entwicklung des Benutzungskomforts kann innovativer sein als jede inhaltliche Neuerung, selbst wenn sie wie ein Rückschritt aussieht.

Es wird der Natur von Innovation nicht gerecht, sie allein auf »die nächste große Idee« zu reduzieren. Es erkennt, dass der größte Teil aller innovativen Leistung nicht in der Entwicklung des komplett Neuen besteht, sondern in der Optimierung des Bestehenden.

Die Kombination von Bekanntem mit Neuem

Das US-Magazin Polygon kürte in einem Artikel die innovativsten kommenden Spiele des Jahres 2014 und bat deren Entwickler um eine Beschreibung, warum ihr jeweiliges Spiel Neuland betritt. Über Sonys Cover-Shooter »The Order: 1886« sagt der Kreativdirektor Ru Weerasuriya: »Der Mix aus echter Geschichte und Fiktion macht unsere Welt so neuartig.« Ryan Barnard von Massive stellt für »Tom Clancy's The Division« heraus, das Team baue »das größte und realistischste Open-World-New-York, das je in einem Spiel erschaffen wurde«. Für Bungies »Destiny« erklärt der Design Lead Lars Bakken: »Wir greifen eine Menge Ansätze von anderen Genres auf und kombinieren sie auf eine Art und Weise, die Spieler neu darüber denken lässt, wie tiefgehend ein actionlastiges First-Person-Spiel sein kann.« Mit anderen Worten: Die entscheidende Innovation dieser kommenden Blockbuster besteht laut ihrer Kreativ-Chefs darin, einen bekannten Kern in ein neues Szenario zu übertragen (»The Order«), ein bekanntes Szenario viel größer zu machen (»The Division«) oder bekannte Dinge originell zu kombinieren (»Destiny«). Es hört sich nicht so an, als ob hier die Zukunft des Spielens neu erfunden werden würde. Aber das bedeutet eben nicht automatisch, dass diesen Multimillionen-Dollar-Wagnissen Innovation abginge. »Call of Duty: Modern Warfare« hat mit der banalen Neuerung, sein Geschehen aus dem zweiten Weltkrieg in die Gegenwart zu verlagern, das Erscheinungsbild seines Genres umgekrempelt. Der Rest des

Spieles war Schema F. Genau darin lag sein Erfolg begründet: Im richtigen Maße innovativ gewesen zu sein, während es den bewährten Rest auf maximales Niveau optimiert.

Unsichtbare Innovation

Es ist auch ein Irrtum, dass Innovation zwangsläufig Veränderung am Produkt sein müsse. Die massive Umwälzung, die die Spielebranche in den letzten zehn Jahren erlebt, passiert beim Geschäftsmodell: Dank Free2Play werden Spiele kostenlos. Das ist Innovation beim Preis. Auch hier sticht Komfort Qualität: Das Konzept ist so wirkmächtig, dass es Mitte der 2000er-Jahre den Vorreiter-Firmen ermöglichte, mit vergleichsweise minderwertigen Spielen berauschende Umsätze zu machen. Inzwischen ist der Markt gereift, und so sehen sich auch Free2Play-Firmen gezwungen, auf traditionellere Arten der Innovation zurückzufallen. Wo eine Differenzierung bei der Produktqualität zunehmend schwieriger wird, gewinnen andere, weniger sichtbare Entwicklungsräume an Bedeutung.

Wie unsichtbare Innovation funktioniert

Beispiel Zielmärkte: Der serbische Entwickler Nordeus eroberte mit seinem Online-Fußballmanager »Top Eleven« den Spitzenplatz in seinem Genre, indem er das Spiel zuerst in Schwellenländern Lateinamerikas und Asiens vermarktete. Dort ist die Spielerschaft zwar weit weniger zahlungskräftig, aber dafür zahlreich und fußballbegeistert – Nordeus hatte in kurzer Zeit Millionen aktive Spieler zusammen und damit die kritische Masse, um »Top Eleven« auch im Westen stark zu positionieren. Die Serben spielten die etablierten Wettbewerber trickreich aus.

Beispiel Vertriebswege: Der Erfolg von Bigpoint entspringt nicht allein dem frühen Einstieg in den Free2Play-Bereich, sondern vor allem der Idee, ein Netzwerk aus Kooperationen mit Medienpartnern zu knüpfen, die den Spielen Sichtbarkeit verschafften im Tausch gegen einen Gewinnanteil. Insbesondere Fernsehsender erwiesen sich als so hervorragende Werbeplattformen, dass Bigpoint seine direkten Konkurrenten in Sachen Reichweite auf Jahre hinaus hinter sich ließ und neue Zielgruppen erschloss.

Beispiel Entwicklungsprozess: Bei gleichwertigen Produkten hat derjenige Anbieter einen Wettbewerbsvorteil, der effizienter herstellen kann. Gerade bei Spielen mit ihren langen Entwicklungszeiten und großen Teams bestehen jede Menge Spielräume zur Ablauf-Optimierung. Rapid Prototyping, Outsourcing, Scrum und Agile mögen (oder sollten) heute Standards sein, waren aber alle einmal Innovationen, die die Flexibilität und Kosteneffizienz des Innovators entscheidend verbessert haben. Solche Prozessinnovationen sind für den Kunden selten sichtbar, aber spürbar durch bessere, günstigere oder früher verfügbare Spiele.



Die Innovation von »Clash of Clans« (oben), »World of Warcraft« (Mitte) und »Candy Crush Saga« (unten) liegt darin, dass sie Bekanntes massentauglich aufbereiten.



Auch die kommenden Blockbuster-Spiele »The Order: 1886« (links), »Tom Clancy's The Division« (Mitte) und »Destiny« (rechts) bauen auf das Konzept, einen bekannten Kern in vollkommen neue Szenarien zu übertragen, bekannte Szenarien stark zu vergrößern oder bekannte Elemente auf neue Weise zu kombinieren.

Erfolg macht Innovation aus

Innovation ist kein Wert an sich. Eine Idee ist bedeutungslos ohne Umsetzung, und die Umsetzung bedeutungslos ohne Akzeptanz. Zur Innovation gehören nicht nur der gute Einfall und dessen Ausgestaltung als Produkt, sondern vor allem auch die Frage, wie man dieses neuartige Konzept dem Publikum vermittelt. Eine Innovation, die erfolglos bleibt, ist keine Innovation. Deshalb kann etwa Ubisofts Julian Gerighty über das Rennspiel »The Crew« sagen, es sei »eine revolutionäre soziale Rennerfahrung, die Autofans rund um die Welt in einer persistenten Online-Welt verbindet«, so als habe es »Test Drive Unlimited« nie gegeben. Aber »Test Drive Unlimited« ist es nicht gelungen, seine Innovation in nachhaltigen Erfolg umzuwandeln.

Innovationen folgen Veränderungen

Innovation ist keine grundlose Veränderung. Sie braucht Grundierung in der Realität. Sicher ist es richtig, dass Innovationen große Veränderungen auslösen können. Aber wichtiger ist die umgekehrte Perspektive: Innovationen folgen Veränderungen. Besonders eindrücklich zeigt sich das in Zeiten der Not, in Wirtschaftskrisen oder bei Rohstoffmangel; nichts treibt etwa die Entwicklung erneuerbarer Energien so an wie ein steigender Ölpreis. Spiele ihrerseits folgen den Geräten und Plattformen, die Menschen im Alltag nutzen. Wer rechtzeitig erkannt hat, dass Internet-Browser, Facebook, Smartphones zu Massenmedien werden würden, die gleichzeitig Unterhaltsbedürfnisse bedienen, der hatte die beste Gelegenheit, seine Produkte für diese Kanäle anzupassen. Jede neue Plattform bringt auch deshalb neue Marktführer hervor, weil etablierten Firmen in der Regel das Verständnis fehlt, dass diese Kanäle ein Innovationsraum sind, der qualitative Rückschritte nicht nur erlaubt, sondern erfordert.

Innovation braucht Zeit

Folgenreiche Veränderungen entstehen nicht über Nacht, sie entwickeln sich. Entsprechend fußt auch der Großteil aller Innovation nicht auf Geistesblitzen oder Bauchgefühl, sondern auf Beobachtung und Analyse. Die wesentliche Strategie, die Peter Drucker in seinem Buch empfiehlt, ist das Hinterfragen von unerwartetem Erfolg oder Misserfolg. Warum ist ausgerechnet »Puzzle & Dragons« so ein großer Hit in Asien? Warum funktioniert es umgekehrt nicht ansatzweise so gut im Westen? Was lässt

sich aus dieser Erkenntnis über die Bedürfnisse der Spieler ableiten? Es lohnt sich, das Verhalten der eigenen Kunden immer dann zu erforschen, wenn es von der Erwartung abweicht. Betreiber von Farm-Spielen konnten den Boom der Casino-Spiele daran vorhersagen, wie gern ihr Publikum am Glücksrad dreht. Aus einem einzelnen Spielelement lässt sich mitunter ein neues Spiel ableiten – wenn man aufmerksam ist.

Nachhaltigkeit ist entscheidend

Umwälzende, alles verändernde Innovationen sind selten. Neuerungen, die radikal mit dem zuvor Gewesenen brechen, bergen ein hohes Risiko des Scheiterns. Auf sie zu setzen erfordert den Mut und die Leidenschaft des Pioniers. Wenn sie Erfolg haben, entfalten sie eine Strahlkraft, die blendend sein kann; der wirkliche Einfluss des Neuen zeigt sich erst, wenn es zum Gewöhnlichen wird. Die Faszination der großen neuen Idee steht nur bedingt im Verhältnis zu ihrer Bedeutung. Die Liste der D.I.C.E.-Preisträger für herausragende Innovation führt das vor Augen. Es ist deshalb erfreulich bescheiden, dass die Jury mit »The Last of Us« diesmal nicht die nächste revolutionäre Spielidee ausgezeichnet hat, sondern ein Spiel, das durch seine ambitionierte Erzählweise Maßstäbe setzt und das exemplarisch steht für den schon greifbaren Trend, in Spielen ernsthaftere Themen auf tiefgründigere Weise zu behandeln.

Das mag keine umwerfende Innovation sein. Aber womöglich eine nachhaltige.

Christian Schmidt



»Test Drive Unlimited« konnte seine Innovation nicht in nachhaltigen Erfolg umwandeln.



Making Games 04/2014

INNOVATING WITH FLASH

The development of Might & Magic Heroes Online issued the challenge to the team to find the right technology. Burkhard Ratheiser explains how the limitations of creating a browser game actually helped to create the game's rendering engine.



Burkhard Ratheiser
is Senior Producer
at Ubisoft Blue Byte.

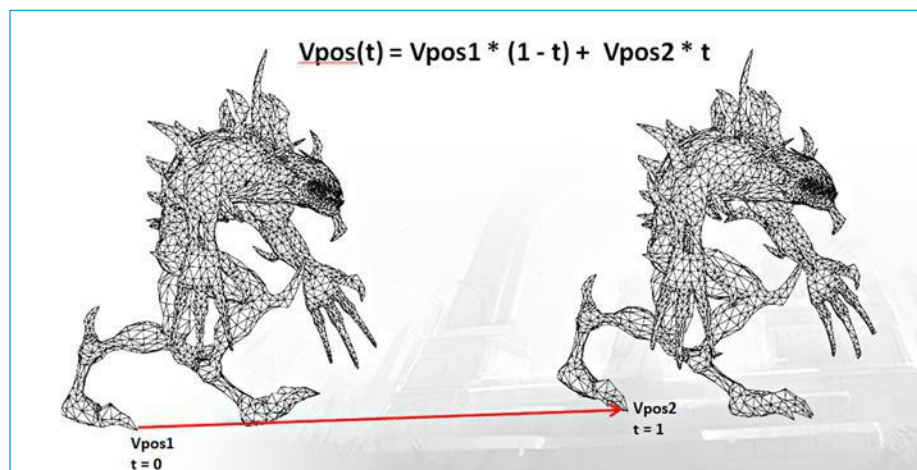
Burkhard co-founded Related Designs in 1995. He received a Diploma in Biology from the Johannes Gutenberg University Mainz. As a Managing Director he was involved in all strategic business decisions, establishing Related Designs to become one of the leading developers for strategy games. As Head of Graphics and Head of Research & Development, Burkhard took care of Related Designs proprietary state of the art in-house technology, ensuring that all Related Designs titles are setting technical and visual standards. The last gaming titles he has worked on have received several awards as well as the renowned red dot award. Related Designs was acquired in April 2013 by Ubisoft. Currently Burkhard is a Senior Producer at Ubisoft Blue Byte, leading a team of outstanding developers.

When we started working on Might & Magic Heroes Online, we had to make a choice in picking the right technology. Since part of the mandate was creating a Free2Play browser based MMO in the »Might & Magic« universe, choosing the right technology couldn't be purely made from a technical standpoint. From a strategic business standpoint it was mandatory to have the lowest possible install barrier for the players to access the game. We were also mandated to create a game with very high graphical production values. For those reasons, we opted out of any technology

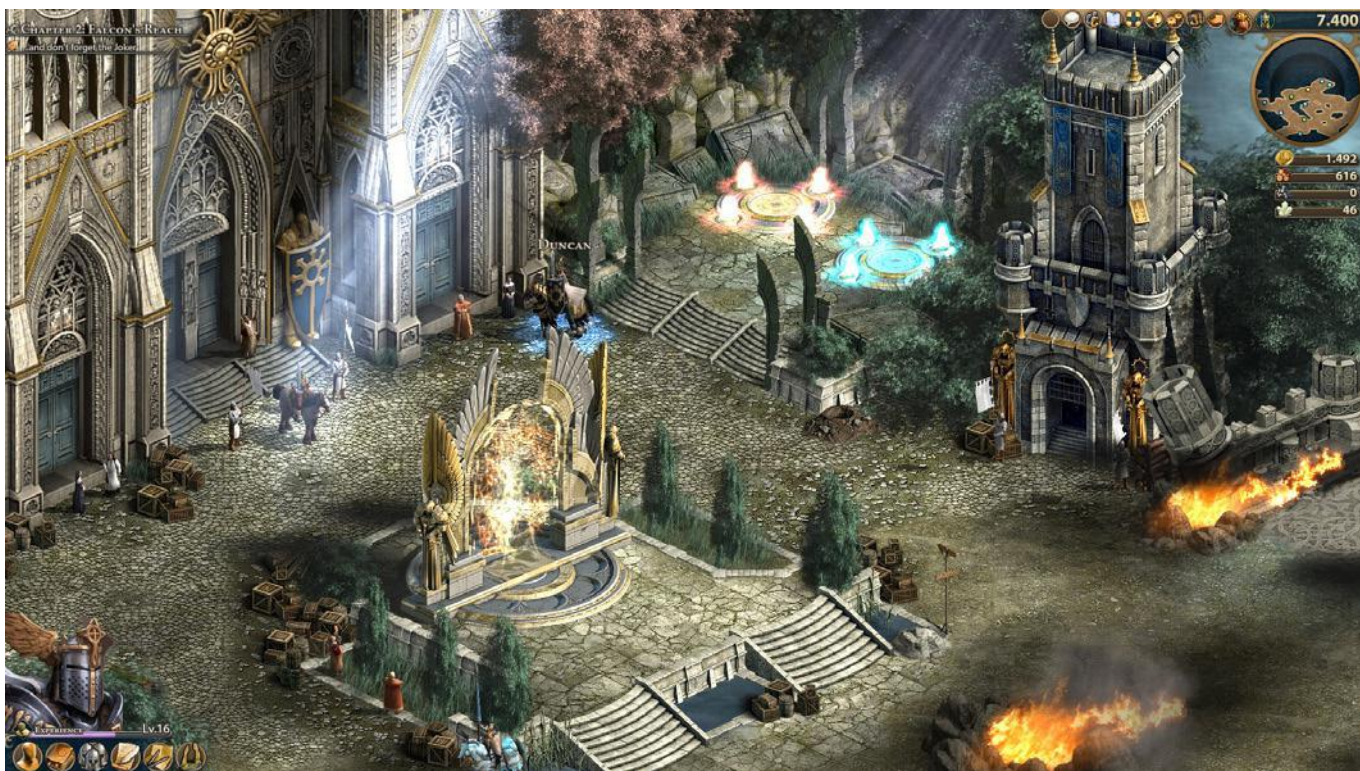
where some sort of plugin installation was being required upfront.

When we started pre-production, HTML5 was being hyped a lot. Still, at the time, due to incomplete and insufficient standards across different OSs and browsers we decided that betting on HTML5 for our production was too risky. So we played it safe by choosing Adobe Flash as a base technology and went into preproduction.

The decision to use Java for implementing the game backend was mainly based on the fact that the backend of »The Settlers Online« and »Anno Online« were also being done in Java. Therefore, we could definitely share some of the code to connect to other backend services from those teams.



Vertices from consecutive key frames are linearly interpolated in the vertex shader.



Might & Magic Heroes Online has very high graphical production values although being based on Flash and without any additional plugin installation upfront.

So, we're doing Flash ...

When we started preproduction of Might & Magic Heroes Online, we first had to educate ourselves with a technology that was unknown to us. All of the software developers in preproduction had previously worked on »Anno 2070« and never worked with Flash and Actionscript3 before. These developers were used to writing multithreaded high performance C++ code and pushing the graphics cards to its utmost limits with state of the art rendering technologies. So it was not only a problem of learning some new, restricted technology, but also coming to grips with some – mostly unknown – limitations.

Flash 2D

At the time we started writing the first lines of Actionscript3 code, we had to rely on Flash's 2D capabilities. So we had to learn some new tricks on how to make best use of those limitations. We were quite lucky in being able to share information with Rainer Reber who had written the 2D rendering code on »The Settlers Online«.

Being a »Might & Magic Heroes« game, we needed to display lots of different units that all have their own set of individual and special animations. As we were required to keep the memory footprint and loading times low, we needed to come up with some tricks to cope with these limitations. To render a 2D animation, it's common practice to have a unique picture of every key frame for its animation sequence. Furthermore, if you want the animation to be fluid and not jerky, you need to have

at least 23 frames per second. Therefore it's pretty obvious that having lots of different units with unique animations will add up to being too much of a burden for the engine.

Molehill

At the same time when we started pre-production, a research project called »Molehill« was in the works at Adobe. The mission of the Flash Molehill technology was to bring hardware accelerated rendering capabilities to the Flash platform. However, it was not part of the official Flash release build and we were unsure when it would be. Whilst working on the 2D rendering part of the engine, we followed very closely Adobe's Molehill project, since we knew that to meet our high quality standards, we needed to switch to hardware accelerated rendering.

Stage3D

It didn't take long for Adobe to announce the renaming of Molehill to Stage3D and that it was to be released as part of the next Flash release builds. We therefore started researching Stage3D and rewriting the 2D rendering part. We already had a fully functioning 2D render engine and pipeline, so in case the release of Stage3D was significantly delayed we were pretty safe with having a fallback.

When we analyzed our first results from rewriting the rendering to support the hardware accelerated rendering, we never looked back again. Hardware acceleration is simply a huge performance boost, even when used »only« for rendering simple 2D sprites.

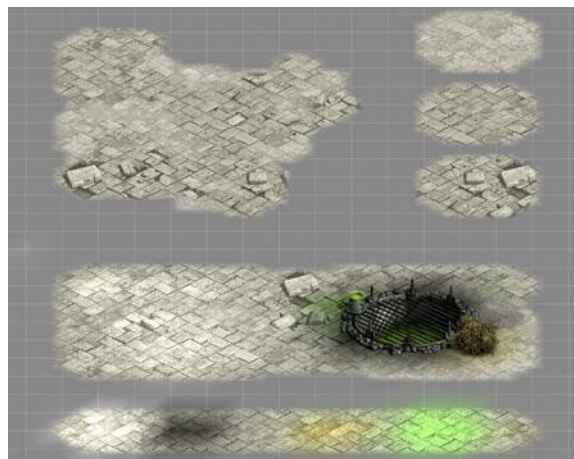
The Game

Might & Magic Heroes Online is the latest entry to the known Might & Magic Heroes series from Ubisoft. Although it is a browser game, it has nearly all known features of its predecessors and expands the universe with MMO elements like cooperative gameplay. Story-wise, MMHO plays between Heroes 5 and Heroes 6 and allows fans as well as new players to play through an exciting adventure set in the world of Ashan. More information at www.heroes-online.com





The rendering of pre-baked and projected real-time shadows.



You can see here the terrain tiles alpha blending and vertex color painting.

Benefits of Hardware Acceleration

Hardware accelerated sprite rendering helped us to display many more variations, since object and image manipulation come almost for free. Making use of fragment and vertex shaders, like we were used to for all our latest games, made all the difference. We added post effects, dynamic lights and real-time shadows. And the best thing was: if done right it used less data than before!

3D Objects & Animations

While the rendering of the 2D animation frames of the units was being significantly accelerated, the memory footprint still remained and still looked too jerky. So we decided to rewrite the unit rendering to use 3D models, which are lit by simple vertex lighting. As for the animation, we had to make a choice between using skinning or key frame morphing. We decided to go with key frame morphing due to its lower CPU footprint, which can easily become a bottleneck when using Actionscript3, accepting the tradeoff of having a bigger memory requirement compared to skinning. Since the units and animations are either being

done with Maya or 3D Studio Max, we wrote plug-ins for these tools that would export the animations into our proprietary format, containing a mesh for every key frame. The key frames were linearly interpolated in the shader to realize absolutely fluid animations. Also the memory footprint was being drastically reduced by requiring one diffuse picture/texture for the units and a much smaller amount of data for the mesh key frames compared to using picture frames.

Shadows

For all the 2D object sprites we have pre-baked their shadows directly into their picture frames. To render the shadows of 3D objects we simply projected them onto the ground plane, shading the respective pixels in the frame buffer. To avoid any overdraw effects caused by overlaying geometry we used the stencil buffer so that any pixel gets darkened only once.

Special Effects

The rendering of special effects also greatly benefited from changing to hardware accelerated rendering. We implemented two different systems for displaying the special effects.

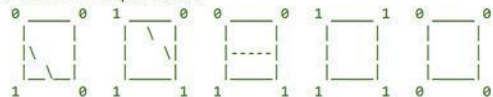
The first solution relies on a 2D sequence of images that are generated by Adobe After Effects and particleIllusion. This is also the initial approach we implemented when still relying on Flash 2D. With the support of the hardware acceleration we can smooth the animations by interpolating the consecutive images in the fragment shader.

The second solution is based on rendering effects being edited inside »3D Studio Max« using »Particle Flow« and exported from there into our proprietary format. This has proved to be a very good choice since we didn't need to devote time and effort to develop an extra effects tool.

We also plan to use this solution for any of our future engines. From a technical point of view these particle animations are based on key frames which are linearly interpolated

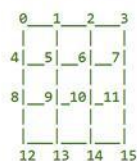
All tiles of the map will be splitted into 9 (3 x 3) parts. Every of these parts can contain one of five possible alpha masks (differently rotated or mirrored).

Five different alpha masks:



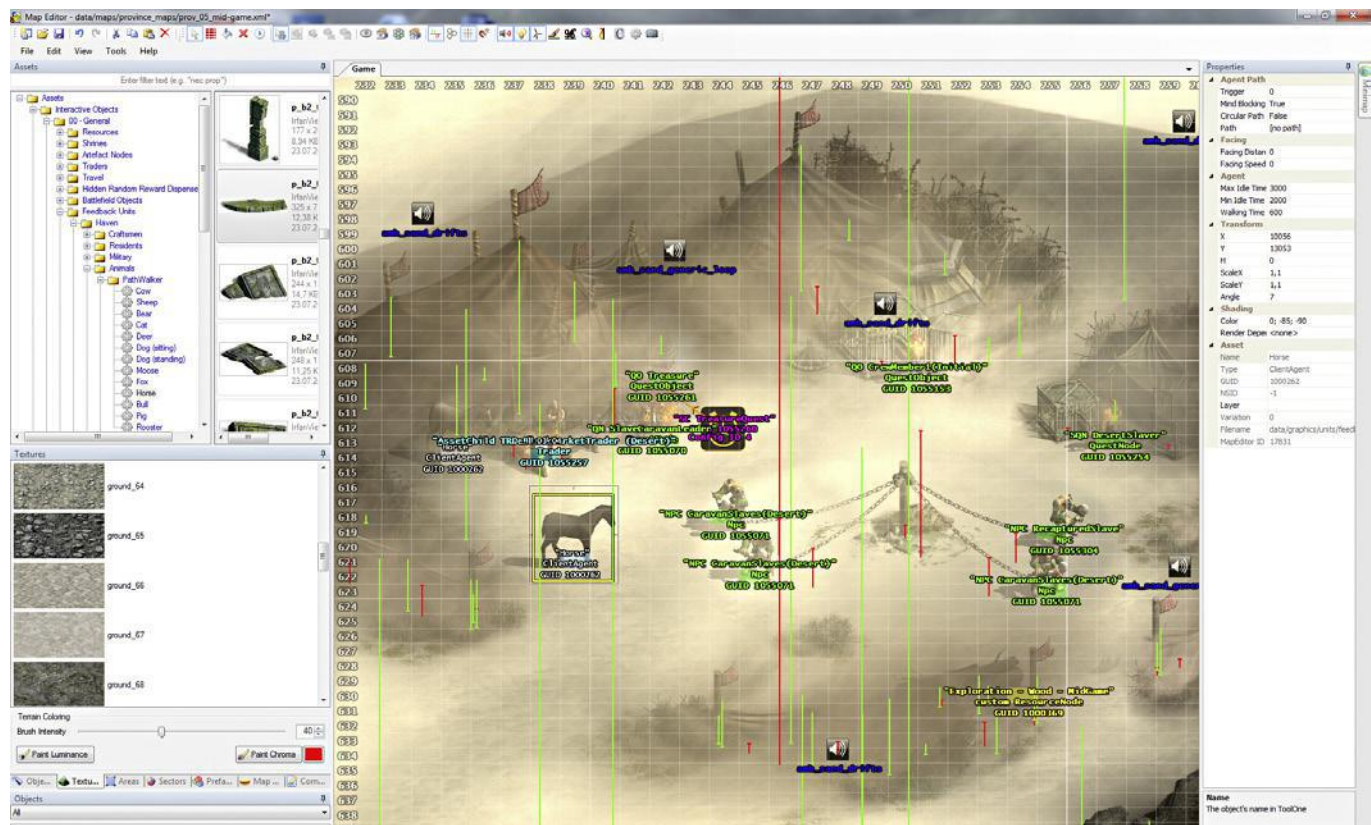
0 = Alpha value of 0 (transparent)
1 = Alpha value of 1 (opaque)

The 16 points of the tile grid will be numbered from top/left to bottom/right, see the image below.



The 4 points (5, 6, 9 and 10) inside the tile have always a alpha value of 1. The surrounding points depending on the type of the surrounding N8 neighbor tiles.

Comments taken from the source code where the alpha masks are generated.



An image from our powerful, yet easy to use Might & Magic Heroes Online Map Editor.

in the GPU to lessen the burden on the CPU. Now with 3D capability at hand, this is the preferred solution as it is much smoother, enabling more variations with the added benefit of a much smaller memory footprint.

Terrain

The Terrain system uses a tile based grid. To break up the tiling effect of the textures, we are using relatively large images. Transitions between neighboring cells are blended by context and stored into an alpha map atlas texture. When saving the terrain map, we store an index into the alpha atlas for every tile and texture. The technique used to render each tile and texture is based on texture splatting, where priorities are assigned to each splat. For further variation of the terrain, we added the capability to paint a color onto each vertex that will be multiplied by the diffuse color.

Water

Being the »Anno« developer, we are experienced in being able to render some pretty cool water effects in 3D. But sadly we were not able to transfer all the fancy render capabilities from Anno 2070 into Might & Magic Heroes Online. With all the restrictions, we ended up with using animated water images that are linearly interpolated in the fragment shader. To display water, the map artists can simply mark certain terrain chunks (a group of terrain tiles) as being invisible so that you can see the water.

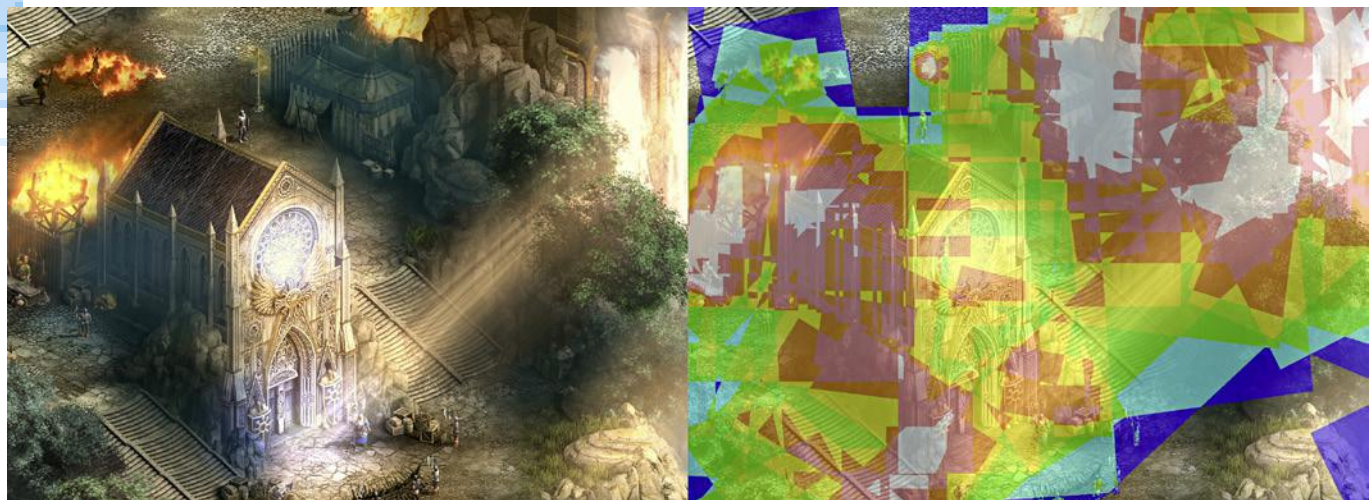
Tools & Content Design

Giving the content designers the capabilities to scale, rotate, mirror and colorize all the assets made the biggest difference. It felt like we were releasing the chains of restriction in which they had been entangled in.

In general we try, where possible, to spend a great deal of time and effort in creating the best content tools possible. So far this has always paid off in developing our games. The creation of content should be as productive and efficient as possible. So for producing all



This picture shows the in-game profiling visualization.



The in-game overdraw heat map. the bright pixels in the right image show areas of high overdraw.

the huge province maps and smaller battle maps we implemented a very powerful, but easy to use map editor.

The programmers implemented all this technology and art did all the graphics but it's the map designers who take these basic ingredients to create something of utter beauty. I also must give special credit to our Art Director, Sebastian Steinberg, who supervised the map creation and pushed the graphical quality by making maximum use of the limitations.

Performance

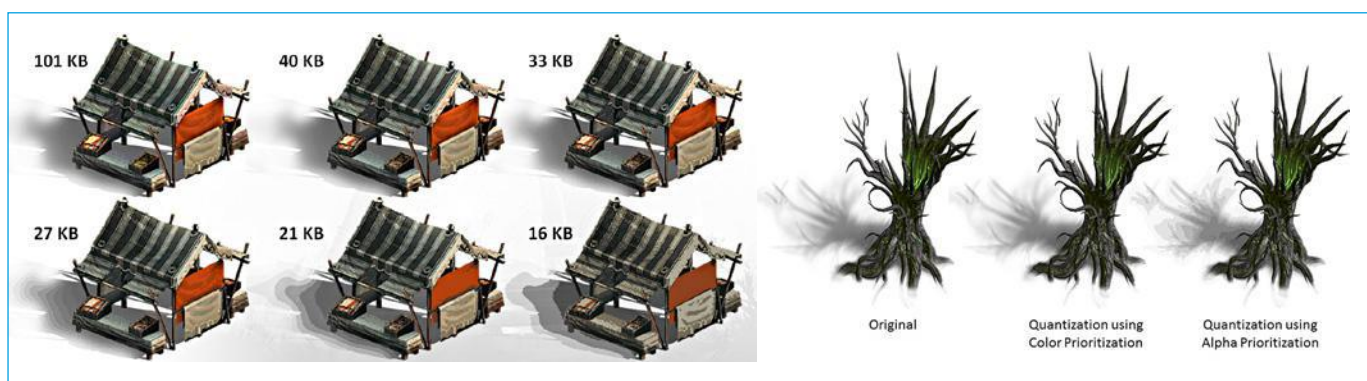
To avoid re-rendering all static parts of each frame and speed up the rendering, we implemented a system called render cache. A render cache is a container of render cache blocks that span a defined screen area and a pool of render targets. Each of those render cache blocks contains a reference to a respective render target with a size of 256x256, which has been proven the optimal size and a reference to all objects inside its area. To initialize a render cache block we assigned a render target from the render target pool. We then rendered all the respective terrain and static background objects into the assigned render target. Later, instead of re-rendering all the terrain and objects again, we simply

rendered the respective render target as a sprite in a single draw call. Any unused render cache render targets will get put back into the render target pool, or in the case of reaching the limit of number of render targets available, will be invalidated.

For a graphic heavy application, memory is always a limiting factor, so we implemented a system called Texture Memory Management Unit (TMMU). All texture allocations are being tracked by the TMMU. If a defined threshold is being reached, the TMMU frees the next best unused texture based on the timestamp of its last usage. If the maximum allowed limit is being reached, then the TMMU forces the release of textures, even when they are in use.

The same principle is used by the Buffer Memory Management Unit (BMMU) for managing the vertex buffers.

Since province maps in Might & Magic Heroes Online are rather huge, it is important to efficiently gather and render only the objects which are currently visible. Therefore we implemented object culling based on a loose grid, which should not be mistaken for a quad tree. The loose grid has several layers with different grid sizes for object with different dimensions. Every grid cell is a container of objects that resides in its own area.



The left image shows the effect when reducing the bit depth by using quantization while the right image shows the result of using different quantization functions.

Optimization & Scalability

Optimization isn't something that should only be done at the end of production. It is of the utmost importance to define the technical limitations at the very beginning of the project. Nor should you turn it into rocket science or simply blind guess. And do not mistake restrictions as being the same as scalability.

Measuring and optimizing the performance is particularly important. To achieve this, we added several visual in-game analytics tools for the programmers, content designers and QA. The process of finding bottlenecks and areas of performance improvements should never be a black box or based on trial and error. Also from a technical point of view we wanted the players to have the best possible gameplay experience.

Since Stage3D rendering is based on DirectX on the Windows platform we made good use of existing analysis tools like PIX, helping us to identify many shader issues. The biggest chunk of our graphical data are the texture atlases that are being used as animated frame sequences and textures. To reduce the loading time we tried several proprietary methods to keep the data as small as possible, but sadly none of those seemed to be worth the effort. Luckily, our main engine developer, Sven Janusch, came up with a solution that really made a difference: the quantization of the images. The idea behind this is to convert the images to indexed color palettes (sub 8bit indices) which reduced their size to about 30 percent. Therefore we developed a proprietary image converter tool that is integrated into our build process. Based on an error metrics driven comparison function it automatically finds the best quality to data size ratio of the images to be converted. For all the other data we simply use ZIP compression.

Especially in the mobile space, there are many computers (notebooks) that still rely on an integrated graphical solution. As these are very much limited by fill rate, we have implemented a way to render the game into a smaller render target that will be later stretched to the frame buffer size. This optimization is also used for rendering the fill rate heavy effects into a smaller off-screen render target which is also rescaled to the frame buffer size in the end.

In order to be able to scale to different hardware configurations we provided the player with different graphical quality options in order to let them adapt a quality to frame rate ratio that best meets their requirements and hardware capabilities.

Evolving the engine

Improving a game engine is a never-ending effort for any online game that wants to stay competitive in the long run. However, major changes of core systems are a nightmare for the quality assurance team since it is virtually



The hero on the left had no customization options since the equipment was already part of its mesh. The equipment on the reworked version on the right was removed from the mesh and replaced by invisible markers that are used to place different equipment sets during runtime.

impossible to test every single aspect of a continually growing game. We therefore focus on changes that add to the existing content without the requirement of extensively altering our massive code base.

We recently made customizable equipment for player characters available. This feature request was added long after we decided to use frame blending animations and since this feature is usually associated with dynamic skinning, things became a little tricky. We basically took our already existing player models and replaced their equipment with marker points that were sampled in each frame. Those marker points were later used within the scene graph to add a variety of different helmets and swords.

The exact position of a player character's hand or head is calculated on the GPU by a vertex shader and is therefore unavailable on the CPU. Hence we need to interpolate the position of the marker points in the exact same way as the GPU does. In conclusion, we were able to link armor and weapons to already existing meshes with only little changes to our code or already existing content.

Conclusion

Coming from AAA HD game development, a key learning for us when developing the rendering engine for MMHO is that you really need to embrace the many limitations and accept them as a challenge. Furthermore, having spent some time working on this project, it makes me realize just how small a part the development of the rendering engine was compared to the rest of the development.

Overall, we did a great job in making the best use of the Flash platform, since many people not involved in the development at the start didn't believe that it's based on Flash.

... did I mention how utterly important the tool pipeline is?

Burkhard Ratheiser





SPOILER ALERT

UNCOMPLETING A GAME



»When everything goes right, go left!« The guys at Megafuzz might have taken that saying a bit too literally when they developed a game you play entirely in reverse.



Jeff Jensen
is Founder, Programmer and
Level Designer at Megafuzz.

Starting at age 14, Jeff spent many hours of his spare time making games, which quickly became his greatest passion. Although he has worked on many hobby projects over the past ten years, »Spoiler Alert« is his first commercial game, which has made him an award-winning developer even before release. He plans to grow Megafuzz rapidly, already having ideas in the works for the next five titles.

Back in August 2011, I attended the IndieTAW game jam in Viborg, Denmark. It was a small jam with around 60 participants and eleven games, including ours. The theme at this jam was »resistance«. After the topic announcement, the participants started discussing various ideas for games based around friction, freedom fighter games about rebels resisting an evil empire, and other very literal interpretations. Meanwhile, I was struggling to come up with something related to the theme that really excited me. My first idea actually came from not having a graphics artist on my team (I was looking at going solo for the jam). Being a programmer, I wouldn't be able to make pretty graphics for the life of me, so I was thinking »resistance against graphics«, as an excuse to make a game that would in no way be based on (good) graphics.

I fondled around with that idea for a while, but it seemed to be a dead end. Then, in sheer frustration of my lack of ability to produce decent graphics, I started thinking about the fact that many games in mainstream video gaming

had to have a nice level of graphics. Mainstream ... that word caught up with me and took me in a completely different direction: suddenly I was thinking »resistance against mainstream in videogaming« – not just in graphics, but in general. And what was the most mainstream thing I could think of in videogaming? That you had to complete a game. I would much rather try to »uncomplete« it!

And from there, the concept pretty quickly came into place; I've always had a profound love for classic platformers like »Super Mario Bros.« and it also made sense to do this »undo the game« concept in a genre that would be as tried and true and familiar as possible. So, the only logical conclusion was to do the game as a blatant parody of »Mario«.

Shortly after I'd begun programming on a prototype, I was approached by this tall guy, who introduced himself as Martin. He could see that I was working solo – and he hadn't been able to find any teammates either. He told me that he was a graphics artist, which just made my day. Now the game's graphics wouldn't have to cause horrible eye bleeding to the players. I told him that my idea was, as shortly put as possible, a »Super Mario« game in rewind. That idea was suitably insane for him, so he was on board. He started making graphics, and I kept on programming.

We were hastily working away, when the arranger of the event approached us. He was making an official list of all the game entries for later, so he wanted to ask for our game's title. But we hadn't made one up at that point. We looked at each other, both looking like giant question marks. I was thinking that, with the game starting off by showing the ending, it would in a sense be a huge spoiler. So I looked at Emil and just flung out: »Spoiler Alert«. He laughed a bit, wrote it down, and continued on.

Why a Chili?!

Showing the game off in different places, like Casual Connect Europe 2014 in Amsterdam,



»Spoiler Alert« is a parody of classic platformers in the sense that you uncomplete the game: you put back coins and revive enemies.

one of the questions we received the most was: »Why is the main character a chili pepper knight?«. Again, I must refer to the game jam. You see, while Martin and I were working on a game and getting to know each other, Martin told me something that quite surprised me: he had never done frame by frame animations before. As that was what I needed, he had to adapt, but we had very limited time to produce the game. So he had to choose a very basic shape that would be as easy as possible to animate. Hence the chili pepper. The crusader helmet was then thrown on, not to make him a knight, but because Martin didn't have time to draw a face. Ironically, the protagonist being a chili with a helmet – two things that we were forced to do out of time restraints – has since become the most iconic thing about the game. Aside from the mechanics, of course.

In the end, Spoiler Alert did well at the game jam. Out of the eleven games, ours won the first prize, an audience award and a judge's favorite award as well. A couple of weeks after the game jam, the national Danish newspaper »Politiken« wrote a very favorable article on the game and its potential.

Martin and I sat down to discuss making the game jam prototype into a full game. We both believed there was still a lot of elements in the classic platformer genre that we could still poke fun at, while trying to breathe some fresh air into a very done type of game. So, we decided to take on the mission of transforming Spoiler Alert from a 48 hour game jam prototype to a full commercial game. A third guy, named Roland La Goy, who had done music and sound for me before on other projects, was added to the project as well to take on soundtrack and sound effect duties.

We were (and still are) all three located in different locations (Martin and I live in different cities in Denmark, and Roland lives in the USA), and we had everyday obligations, so the progress was slow (only in our spare time), and all communications went over internet or phone. Occasionally, Martin and I met physically to flesh out some more advanced things in the game, but 90 per cent of all the game work was done over the next 18+ months via virtual communications. Facebook, Skype and Google Docs were tools we used very often. Oh, and Dropbox. Gotta love that one.

Thinking Backwards (Was Harder Than We Thought)

Now, this was very new to both of us. I had only done a few hobby game projects before, and Martin had never been involved in game development at all. And to add to that, not only were we both inexperienced, but we had also chosen a very tricky game concept. Designing a game like Spoiler Alert that goes completely backwards presents very unique challenges. Not to mention that even wrapping our own heads around the concept was actually dif-

ficult enough in itself. One instance of this I remember specifically well was the first boss fight we had to implement. It was a boss that was already dead when the level began, then you had to revive him, and then he would fly around in a reverse path and swallow all the fireballs he had thrown at you. However, with the »right-to-left« way being so ingrained our brains, we ended up making the boss fight like a »normal« boss fight. What this means is that the boss was already alive when you met him, he was flying around in a forward-going path, and he was shooting fireballs at you – not having them launched back into himself.

This was so engrained in us that we had both been looking at it for about ten minutes, and declaring us both satisfied with it – only then to realize we had done it wrong (or right, depending on your perception of course). So there we were, two developers who had just done an entire boss fight ... the »right« way, and staring at it for several minutes before realizing the mistake.

To me, that was the first time I really realized how aware we had to be about such things, and really, really force our brains to work in reverse. This may all sound very simple and straight-forward to you, but you'd be surprised at how ingrained the normal forward-going thinking is. At least ours was.

Other unique design challenges were very much revolved around designing gameplay elements that worked very well in a traditional game as well when turned 180 degrees. Some of them were easy enough: spike pitfalls work the same, coins dictate a set path (avoid the coins that were not previously collected and put back the ones that were) and so on. But other elements, like falling spikes (the ones that you usually run past before they hit your character in the head) presented a unique problem. If they worked like in a traditional platformer game, they wouldn't be a challenge; they would just fall back up into place and never pose a risk of killing you. But, being a traditional platform element, we wanted them in the game.



Made in roughly twelve hours, this is what Spoiler Alert looked like at its infant stage during the game jam.



In the first design of the boss fight, the »Big Bad Squid« was alive when you met him and shooting fireballs at you – the wrong way around.



Spike pitfalls (upper image) work the same even played in reverse. However, another traditional platform element presented a problem: falling spikes (lower image) wouldn't pose a risk if they worked like in a traditional game and just fell back into place.



The original version of Spoiler Alert (upper image) was taken into full re-development and received a major graphics overhaul (lower image). New levels were added and the existing ones were cut into smaller pieces to make the game more mobile friendly.

At first, we experimented with having them move at different speeds; some you would have to run past, others you would have to jump over. But this presented two new problems; first of all, it didn't make any sense in terms of the concept, and second of all, it wasn't very intuitive to the player.

In the end, what we did was to mix them in with regular spikes. Because the falling spikes would already be on the ground when you approached them, and because they used the same sprites as regular spikes (only vertically flipped), they would be visually very similar when mixed tightly together. This way, the falling spikes were easy because you just had to run underneath them when they were falling back up into place, but at the same time, it forced the player to be observant on the surroundings since some of the spikes on the ground were the regular types that you had to jump across.

Make a Game, Throw it Out, Then Repeat

After about eight months of development, we had (what we at that point considered) a complete game. We were about ready to release the game on Windows, Mac OS X, Linux, iOS and Android. Now these things were going in a more business related direction, it was time to register an official company, sign some contracts and other grown-up stuff. Yes, I know, in retrospect, this should have been done way earlier. You live, you learn.

So around this time, I was doing a lot of paperwork. I was registering the company, talking with a lawyer, making sure that our company name would show up on the App-Store and other places (which apparently was way more difficult than I would have thought) and a lot of similar stuff. While all this was going on, we couldn't really release the game, and this forced us to take a step back from development and look more objectively at the game.

We discovered that what we had done was simply not to our satisfaction. We knew that we could do so much better!

So, not being dependent on the imminent commercial release of a game (because we were still doing this in our spare time out of pure passion), we agreed to postpone the release to take the game into full re-development. So many things were improved.

First of all, the game received a major overhaul in the graphics department. Everything visual was improved; the colors, the animations, the amount (and quality) of terrain content; just about everything. Secondly, the game content's quantity and quality was also massively improved. We especially did a lot of work on making the game way more balanced in terms of difficulty, we improved the tutorial messages, and we did massive work on explaining the game concept better and faster, as well as making the game way more intuitive and user friendly in general.

Also, to make the game more mobile friendly, we decided on using a snack-sized level design, so basically cutting a lot of the levels into smaller, more numerous pieces. In addition, many new levels were added, because there was not enough gameplay content; the game could be beat in just about 20 minutes. So we went from having 30 fairly long levels to 100 levels that were a bit shorter. In the end, the game time went from about 20 minutes to 60 minutes (not including any replay elements like scores, records, achievements and hidden objects).

This extended the development time a lot more, and in the end, what started out in August 2011 was now well into January 2014. The game was received quite well (especially when you consider we hadn't published as much as a demo or done much to promote the game), like the aforementioned article in »Politiken«, as well as »YoYoGames« (the creators of GameMaker: Studio) having reached out to us because they wanted to feature our game in their official games showcase – along with such great names as »Hotline Miami« or »Super Crate Box«. There were other small things like that, and they were great for morale, but we were exhausted from working on and looking at the same game for so long. Friends and family were starting to ask if the game would ever be coming out, and a big part of us just wanted to get it over with and get it out. But we just kept on refusing to release something that we knew we could do better.

Everything happens for a reason.

Wait and You Shall Receive

In January 2014, we were lucky enough to be invited to the Casual Connect Europe 2014 in Amsterdam to show our game among more than 110 other developers in the Indie Prize Showcase. This was the first time we would ever get to show off Spoiler Alert at a big convention, so I took off with postcards, stickers and a big banner to promote the game.

We were thrilled just because of the opportunity to show the game and possibly meet some interesting people, but did not for one second even think about the Indie Prize award, because we didn't believe our game was up to the competition. We were then overwhelmingly surprised when Spoiler Alert (as the only game there) was nominated for three different prizes: »Most Innovative Game«, »Best Desktop Game«, and »Most Promising Game In Development« (which we won). At the convention, we also got to speak with interested publishers, one of which we ended up signing with a few weeks later – with a deal to publish the game on Steam, iOS and Android.

This was huge to us – we had previously tried getting onto Good Old Games, but they kindly rejected us, saying that the game was not suitable to their catalogue. Being rejected from GOG, we didn't even stop to think about pursuing the dream of Steam, and suddenly, we had



The graphical assets - ranging from terrain content to enemies and bosses - of Spoiler Alert were handled in GameMaker: Studio.

signed a contract that would make sure that we got onto Valve's platform. It was surreal. And it was also lucky; if we had published on our own just a month before as initially planned, this would never have happened. So sometimes it does pay to delay. At least if you're in the lucky position of a choice in the matter.

Conclusion and Looking Forward

As of March 2014, we are currently doing some final adjustments on Spoiler Alert before the release on Steam and other major platforms in June 2014. First of all, we are adding a level editor which will work with Steam Workshop so you can create levels and share them with everyone - and you can download and play theirs too. We are also adding a hardcore speedrunner mode to appeal to the more hardcore gamer. Additionally, we integrate the game with Steam, to make sure that achievements, scoreboards etc. will work as expected there too.

But in many respects, the game is done, and we are standing before a very close release. So, looking back, what have we learned?

Well, I think that this being our first »real« game, it has taught us a lot of things. I know it has for me, at least. First of all, it has taught me a lot about how much the small things that

make a game polished matter. In the past, I was focused solely on making the game, not making it pretty, fluent or polished. But the small things are such a huge part of a game.

Secondly, I've learned a lot about working with a team. And as an extension of that, I've learned much about the importance of trying to work on a realistic schedule (and not to underestimate the value of realistic timeframes! If you think a task takes a week, try adding one or two more to that).

I've also learned the importance of networking. I've always known this somewhere in my heart, but never truly felt the power of networking until somewhere in the process of this project. Talking to Julie Heyde, a gaming agent and game developer herself, who introduced me to Yulia Vakhrusheva (the woman behind the Indie Prize event), was what got us to a point where we could be nominated and win prizes - which would make publishers interested in us. It gave us an incredible exposure, and none of this would have been possible without some simple, but very powerful networking. And amazingly helpful people. Seriously, if any aspiring game developers are reading this - I know that hiding behind your keyboard in the bedroom is tempting - but network, network, network. And then network some more (not the WiFi kind).

And lastly: I've always been a fan of originality, and a lot of the people who were interested in Spoiler Alert have been because of the unique twist, which is great. In that regard, it has paid off to live by the old saying »When everyone goes right, go left!«

However, next time, I will try not to punish myself or my poor teammates by doing another ridiculous game concept like this. It just hurts the brain too much.

Jeff Jensen



A very early version of the level editor that will work with the Steam Workshop and will allow users to design levels for other players.



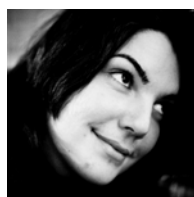
A big banner, postcards and stickers were used to promote Spoiler Alert at the Casual Connect Europe 2014.





QA FOR INNOVATIONS AT BIOWARE

Building complex, innovative open-world-games like »Dragon Age Inquisition« calls for an innovative QA department. Tulay McNally explains how QA is all about preventing failure and why knowledge sharing is essential for every studio.



Tulay McNally is Director of Studio Development QA and Quality Engineering Group at BioWare Canada.

Before joining BioWare in 2010, Tulay has worked for companies like Microsoft, Alias Systems, CG Channel, Square Enix and SEGA. As Director of Development QA and Quality Engineering, for BioWare Edmonton & Montreal studios, Tulay is responsible for strategic planning, organizing, directing and coaching the QA teams with a focus on QA Managers and Leads. In conjunction with the other studio department and project leaders, her goal is to ensure the success of all projects at BioWare Canada.

[@Tulaychan](#)

When I started to work at BioWare Canada I thought it was the best company I had ever worked for. Almost four years later I still do. Our studio culture reflects our commitment to quality. Every development discipline cares about quality. This is an important pre-requisite to understand that QA is not the only department that assures quality. QA does not own quality and they are not the gatekeeper of quality.

Common misconceptions about Quality Assurance

There is that notion that QA is the »last line of defense« – the gate between developers and the customer. In the traditional Scrummerfall scenario, the combination of Scrum and Waterfall, QA often comes online during Alpha or later. Often seen in this scenario is also the exclusion of QA from the development process and testing is only allowed or performed at the end of a sprint. There is also the common misconception that the purpose of QA or testing is to find bugs. If you think so, too, you might be in for a surprise on the next few pages. So please, read on ...

White Box vs. Black Box Testing

In an Agile team the test or QA team is embedded. In the game development world this means we do what is mostly referred to as Developer QA or White Box/Grey Box testing. Before I dive a bit deeper into our organizational structure and agile testing, I would like to briefly go over the differences between QA and, well, QA. Or rather: the differences between White Box / Black Box Testing and Quality Control versus Quality Assurance.

White Box testing takes an internal perspective of the system to design test cases based on the internal structure of software. White Box testers often understand code, have a deep understanding how the systems interact and understand art tools, pipelines, editors etc. Generally speaking: they don't have to know how to code or be an artist, but they have to understand the process, dependencies and how things work in general, so they can reverse engineer them, exploit systems, find their weaknesses. Black box testers often have minimum knowledge of how the game is created. A big difference, compared to White box/embedded testers, is that they typically don't have access to the same tools developers use. They pretty much test the game like an end-user would.

Quality Control vs. Quality Assurance

QC (Quality Control) is a failure detection system that uses testing techniques to identify errors in products that tests the end product in intervals, to ensure the products or services meet the requirements as defined earlier during the QA process. QC is almost like a filter at the end of various stages of a project, testing pieces as they are finished to ensure nothing buggy or substandard made it in.

WHITE BOX TESTING VS. BLACK BOX TESTING

Also known as glass box or transparent box testing	Testers have typically no knowledge of the test objects internal structure
White Box testers are an advocate for the end-user and are typically embedded with developers long before pre-alpha	Black Box testers are often referred to as the »Standard Testers« (or publisher QA) who interact with the game like the end-user would
Done by dev-testers and also by developers	Also known as data-driven/functional testing

While White Box testers have a deep knowledge of the internal workings of game development and even understand code, while Black Box testers usually have no knowledge of internal structures and rather test the game like an end-user would.

QA on the other hand is a failure prevention system that predicts almost everything about a products safety and quality standards. It tests everything that could possibly go wrong and allows you to take steps to control and prevent flawed products or services from reaching advanced stages of product development. Quality Assurance acts as a quality gauge during development, testing not only the product, but the processes used to create the product. QA is intended to be pro-active and catch problems BEFORE they make it into the product.

Becoming Agile is Hard Work ...

... and you may have to deal with resistance to establish change. Our studio has been practicing agile for almost ten years, with QA being closely embedded – part of the fabric of development so to say. We’ve tried different models and keep refining them, since one of our core values at the studio is that we’re only as good as the last game we shipped. Doing Scrum or XP or anything agile is hard and requires a lot of discipline and trust. QA at BioWare had time to mature and refine its processes over the course of a decade – and that is the foundation for a healthy organization and successful collaboration for developing high quality products.

Every Developer Assures Quality

Our developers address quality in each sprint, for example when we have dedicated bug-weeks. These happen at the end of every sprint to ensure what we show-off during our sprint review is actually going to work.

Some of the typical Quality practices of Developers would be activities like:

- design and art peer reviews
- regular triages that keep the technical debt manageable
- daily playthroughs by the core leadership team and internal focus tests
- code reviews
- preflight system

Teams that blame QA for not finding certain bugs still live in the dark ages of development because they don’t understand that quality (or the lack of it) is the responsibility of the ENTIRE development team and that QA is only one of the groups contributing to this effort.

One of the roles of QA is to lead the road to quality by influencing the whole team to put quality at a higher priority and by making sure the right tools and processes are in place to enforce quality through them.

Software Quality vs. Game Quality

Quality has three components:

- Quality of the Software (general functionality, code quality etc.)
- Quality of the Game (content, art, design etc.)
- Quality of the User Experience & Eco-System (is the game fun, is the online experience cohesive etc.)



Black Box testers interact with games like »Dragon Age Inquisition« like the customer would.

QUALITY CONTROL VS. QUALITY ASSURANCE

Makes sure what you have done is what you expected.	Makes sure you are doing the right thing.
Focus on defect detection.	Focus on defect prevention.
Black Box.	White Box/Grey Box.
Example: Compliance Testing, BVT (automated build verification testing).	Example: scrum support, qualitative feedback.

Quality Control is responsible for identifying defects while Quality Assurance focuses on preventing failures.

A typical BioWare RPG is engine light but very content heavy. About 70 per cent of our issues are made up of content issues (Cine/Level Design, Art & Animation).

Therefore functional software is only one aspect of our effort. Stable core systems and tools and our generally mature software quality practices enable our content creators to build those immersive worlds that our fans love.

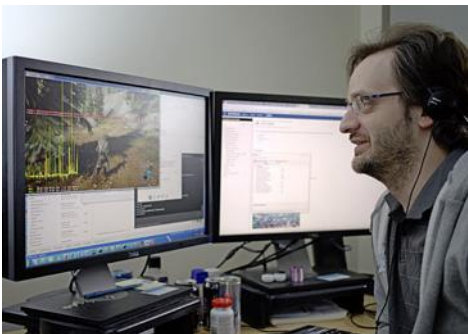
To give you an example: our recent title »Dragon Age: Inquisition« is:

- 4-5 times the size of Ferelden (which was the setting of »Dragon Age: Origins«)
- built with a new engine (Frostbite)
- open world
- ships on current- and next-gen platforms
- has a social integration (<https://dragonage-keep.com>)
- multi-language full VO

It becomes easier to manage this level of complexity with a mature and consistent QA team. QA can identify bugs in the code base and report 2-3 times more valid bugs due to their intimate knowledge of the game and systems, which improves developer velocity. An early analysis of planned or implemented features can help QA to focus more on the customer experience and report potential implications of issues to the end-user AND development early.

QA needs to be holistic; it’s not just about making the game functional. Who cares if the software works, but the game is not fun or looks and plays awful?

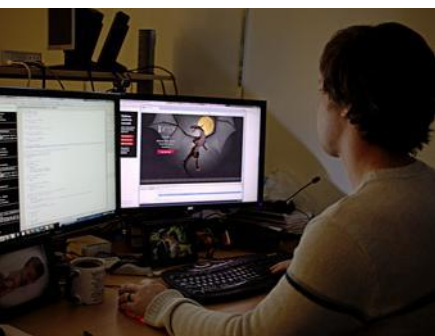
QA at our studio always happens in support of the titles and the studio’s vision. It’s always a context-based strategy; there is no cookie-cutter model. The team is dedicated to the success of the game. When QA is invested in the same purpose as the rest of the development team,



Quality Assurance Lead Darren Clark (upper image) is testing a very early version of a level of »Dragon Age Inquisition«. Software Development Engineer in Test Allan Schumacher (lower image) is checking the CPU and GPU data. (Photos by Alex Lucas)



»Dragon Age Inquisition« is 4-5 times the size of the first game's setting. This called for stable QA tools and core systems.



»Dragon Age Inquisition« also has a social integration, which is tested by Brian Mills, Senior QA Analyst Online.

working on a game becomes more than a job. At BioWare, QA is also a career and not just a job, our dual-career progression path and the pay reflect that maturity as well. We focus on continuous improvement, discipline excellence and knowledge sharing to produce great coherence, which is empowered by our studio culture.

In our organization, testing is involved in the development process from the outset – from vision and concept over to pre-production, full-production, finaling and post-release support.

Quality Management is done at every point in the game development cycle. At BioWare, we involve QA early on to consult the project teams on the testability of features, identify things that might be hard to (or impossible to) test, help develop test approaches in advance of anything getting built, raising potential risks and providing feedback from a player perspective (representing all different player personalities).

At BioWare, QA means integrity, problem solving and ... fun!

Integrity, because first and foremost we're honest. People make software, people make mistakes (including QA); we accept those mistakes and work to improve them. And in the end, we're proud of what we do, because we know we're doing it to the best of our ability.

Problem solving, because in an ever-changing environment, we continually encounter new problems. We also investigate old ones, finding better solutions to prevent them and create better tools in order to create better products.

Last but not least we don't forget to have **fun**. This means more than goofing around for a few minutes in the morning meeting. This is about having a positive attitude about the work we do. A necessary part of problem-solving is identifying problems; it's also about discovering opportunities. Our QA are excited to be part of a collaborative group who are always looking for new improvements.

The earlier QA is involved, the sooner they can start reducing risks to development and for the

end-user. I like the analogy of comparing testers to Military Intelligence, leading people away from seeing testers as gatekeepers (QC vs. QA).

In many ways, the objective of a tester and an Intelligence Officer are very similar. Testers are tasked with providing information to developers and production to do their work better and to make the correct decisions. Just like the Intelligence Officer is not tasked with fighting the enemy in close combat, testers are not tasked with writing the code or fixing the bug.

An agile QA team helps developer on the front lines. Not being a developer doesn't mean that QA doesn't have an intricate and highly technical job. QA often has to think about and simulate strange but realistic scenarios where customers will be using our product in unforeseen ways.

QA often has to work with incomplete information, making assumptions and explaining risks that may or may not happen. Our QA Analysts have mastered knowledge of a product's systems, its content and tech goals and report on both the functionality and the subjective quality of the product, enabling the rest of the team to make decisions with the best information possible. Predictability is important to guarantee success, especially during production. In theory, you should be able to ship a quality game earlier the more knowledge and experience you have. Therefore, knowledge is the greatest asset your studio can create!

Agile Development with Scrum and Agile QA

Agile development focuses on building knowledge about value, cost, and schedule and adjusting the plan continuously to match reality. In traditional software development, you would normally define requirements, design them, implement them, then verify/test everything and release the software. In traditional games QA, as mentioned earlier, you would have a rather small team of developer QA (if at all) and have a larger group of testers probably around alpha. At this point, it can be very costly or impossible to make any bigger changes if they are caught by the testers. At this point, it is also difficult to call this method QA. It falls more under the category of Quality Control, where you are making sure that the game is functional and meets certain criteria in order to be able to be published. Of course, and thankfully, this is not the case with every developer, but there are still many out there who follow this approach.

We don't follow a rigid »one size fits all« operational model – we rather follow the agile testing framework within our context. We generally divide our department into three specializations responsible for different aspects of the game and tools (Content QA, Tech QA, Test Operations). The structure is agile, no team is rigid – every team fine-tunes its structure to suit each project's needs. During prototyping and pre-production, our analysts are mostly involved

in the verification of systems since there are not yet many assets to test. These are typically testers that we refer to as »Tech QA« who have an understanding of the systems that are used to implement the content. They identify issues in the tools, systems and workflows and mostly support the programming teams.

We also have analysts that have an expertise in what we call Content QA, focusing mostly on Level Art, Level Design, Cine Design, Gameplay and Narrative. QA Analysts are often embedded with the scrum teams, with one analyst sometimes supporting multiple scrums. As the project starts to progress and more content comes in to test while more platforms start coming online, the growth and complexity in development often requires more testing of added features and developer tools; more regression testing is required to ensure previously added features have not broken anything.

We made the decision that at this stage of development it is more efficient to move QA into a pool system to even out the workflow. In the current development of our products, we are actually experimenting with different models of embed and pool systems, different team set-ups and techniques such as Session Based or Context-Driven Testing. Technical QA focuses mostly on the systems that are used to implement the content. Analysts also identify issues in the tools that developers are using and flag

issues with systems and workflows. The team would primarily support programmers and anyone who is using our toolsets which would include artists and designers indirectly, as well. Tech QA are responsible for verifying developer tools such as Unreal Editor, Designer Toolset, Story Manager, as fixes and changes are made.

The Tech QA Team works on each system as they come online with the goal of ensuring that it works correctly (incl. the tools), before adding the next layer of a system.

Some of the things that Tech QA would typically cover are:

- Creating test content to help ensure system stability



QA testers are involved early to help spot potential risks and to provide feedback from a player perspective.

NO BROKEN BUILDS : INCREASING TEAM VELOCITY WITH PREFLIGHT

THE PROBLEM

Builds continually break due to code not compiling, slowing team productivity.

- ◆ People can't work
- ◆ It takes too long to iterate
- ◆ New work doesn't get into the game

THE SOLUTION

Assure quality beforehand rather than only measuring it afterward: create a system to automatically test code **before** it goes into the game.

HOW IT WORKS

- ◆ Programmers submit changes to Preflight, not source control
- ◆ Preflight tests changes against HEAD branch
- ◆ Changes that pass are automatically committed to source control
- ◆ Changes that fail are blocked and sent back to the submitter with details

REGULAR BUILD SYSTEM

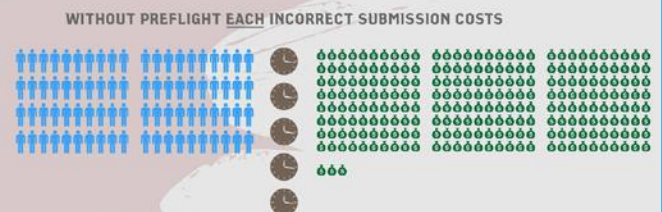


BUILD SYSTEM WITH PREFLIGHT



RESULTS

1 PREVENTING ERRORS SOONER DRASTICALLY REDUCES THEIR COST



2 FEWER BROKEN BUILDS MEANS PAYING FOR THEM LESS OFTEN



Build Failures on Dragon Age: Origins

3 FASTER TESTING INCREASES DEVELOPER VELOCITY



Dave Schaefer Software Developer dschaefer@bioware.com Warren Heise Art & Design

EA BIOWARE

This chart shows the development process of the QA tool »Preflight« that is used to prevent broken builds. As a result, costs are reduced and developer velocity is increased.



Jason Leong, QA Manager for Mass Effect 3, and QA Analyst James Leung are analyzing spreadsheets.

Technical and Content QA

Technical QA (programming support)

This team focuses on ensuring that the dev-team is unblocked and also ensures that the product is functional.

- Test Focus:
 1. daily BVT – to ensure content creators can continue working
 2. Scrum Acceptance Testing
 3. reporting on system status
 4. test script creation and maintenance
 5. Regression Testing
 6. general Ad-hoc and Chaos Testing
- Our technical QA Analysts need to understand the process and logic of systems that enables them to go beyond basic fail/pass cases, but theorize on possible fail cases

Content QA (art, design, writing support)

- Broken up in 4 main areas:
- **Gameplay QA/Level Design:** assists developers create well-paced, fun and balanced games; test functionality and flow of levels
- **GUI:** journal, codex entries, character creation (test for UX and functionality flow etc.)
- **Narrative:** provide feedback about narrative experience from various end-user POV. Test narrative elements, look for IP consistency, test story/VO/ cinematics)
- **Art QA:** artists, tech art, asset verification, asset testing, build test content for developers

Technical QA Analysts focus on the functionality of the product while Content QA provides feedback about the narrative experience or even builds test content for the developers.

- Ensuring tool stability, so that the project can continue making more systems
- Implementation of test hooks for the debug console
- Writing automated scripts against fragile systems
- Helping Content QA members and educating them on how the game functions, so they can perform deeper testing and support

Around beta, the team switches to »end-user-mode« and uses their knowledge for ad-hoc testing. Content QA focuses on what is actually implemented in the game and verify each layer of content. During pre-production, the Content Analysts support the content creators (Level Art, Level Design, Narrative Design, Cine Design, Gameplay Design) and give feedback on the systems and raise risks on content documentation prior to development time being invested in them. During the Finaling Stage, the team switches to »end-user-mode« and stress the product as much as possible until the desired user experience and stability have been found.

Some of the things that Content QA would typically cover are:

- Testing new and existing content for continuity and stability
- Level review processes, which result in smoother review meetings and allows QA to point out concerns that would severely impact the effectiveness of the reviews
- State of the game analysis – since QA has no »stake« in any particular development discipline our analysis is less likely to be unbiased
- Content QA also acts as an advocate of the end-user. QA can hold the dev-team accountable for what the end user might actually see, instead of what the designer thinks users will see
- Content QA is heavily involved in all aspects of design, art, audio, and writing

Content QA Analysts review most plans and documents for levels, creatures, powers etc. and provide feedback and suggestions as well as attending planning and development meetings. Content QA runs acceptance tests on the different level milestones to ensure areas are appropriately hitting their goals for all departments. They also perform focus tests, playthroughs, and whatever other directed or chaos/ad-hoc testing a developer might need.

And the Moral of this Article is ...

... that there is just too much to talk about on the subject of the different types of supports and specializations we have for our studio (in the context of RPG-QA) that I could probably fill an entire book. QA is not just about »assuring quality« – and I very much doubt that one discipline can do that or can be held accountable for the quality of a product. Quality is owned by everyone who is creating the game.

For me, it's very important that QA is a development partner and discipline. I don't want to have a quality police enforcing standards. I'm lucky that at BioWare, our developers already have high standards and practices. If I had to summarize what BioWare QA is in one sentence, I would say: We provide developers with the information they need to make the best decisions about a product and to gather and share knowledge on the state of development.

We look at how the project overall can be more effective. Our focus is on the prevention of defects and being pro-active about it. We don't just say: »here is the problem«. We say: »here is the problem and this is how you can fix it or at least make it better.« The more transparent and clear the feedback is, the better it helps development to react more effectively.

QA is always context-based. The focus on methodologies of each of our games changes from game to game across our IP's. Having historical context of the development of past projects and their QA strategy helps to avoid common errors when developing new models. That's why staff retention can be important.

On Dragon Age, for example, the QA Manager is part of the Core Leads group. Aside from creating the QA strategy in support of the game's vision, the QA Manager also reports on the state of the game, makes an early risk assessment before the vision even reaches the developers. He gives early feedback on design decisions with a historical context. Our QA effort-breakdown reflects the areas in the game they focus on. We generally speak of Content QA and Technical QA.

QA develops test content of a wide variety that can be used, not just for testing, but also as a tool to be used by development for example for reviewing assets, verifying systems/content and profiling, among other things.

This can sometimes lead down to a dangerous path as well, when the lines between what is Quality Assurance vs. Developer Support get too blurry. Where and when do you draw the line? The work needs to get done, right? Developer X needs help. QA Analyst X is intimately familiar with the editor, why can't he just fix the bug, right? Embed QA has its pitfalls as well, but that's a different topic – a great topic for a future article, in fact. As Leaders in the Games Industry it is our job to foster a healthy culture. Leaders take risks, leaders aim high and leaders stay team focused and are persistent and flexible.

Some people fear change, but change is only bad if you fail to evolve. Our industry is one of the most dynamic industries. If there is one constant in game development, it is change.

Remember that the backbone of success of any organization is team-work and collaboration. And don't forget your people. Smart, passionate and hard-working people can create amazing things. Together, we can bring about a long due change in our development culture and our relationship with QA.

Tulay McNally



DIE ERFOLGREICHSTE
TALENTBÖRSE FÜR
SPIELE-ENTWICKLER

POWERED BY



DEIN WEG IN DIE SPIELEBRANCHE!



**Mehr als 300 Stellen haben wir
schon vermittelt. Ergreif jetzt
deine Chance auf einen Traumjob!**

19. Juli 2014

IHK Frankfurt

Anmeldeinformationen unter
www.makinggames.de/talents

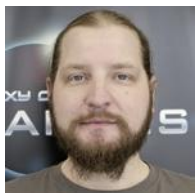




Making Games 04/2014

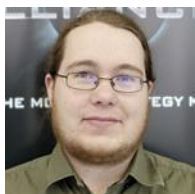
DAS BACKEND DEVELOPMENT VON GALAXY ON FIRE - ALLIANCES

Nicht alle Innovationen sind sichtbar. Für ihr erstes Free2Play-MMO musste Deep Silver FISHLABS ein komplett neues Backend-System auf die Beine stellen. Eine gewaltige Herausforderung, die wie jede Innovation eben nicht nur gute Ideen erfordert, sondern vor allem praxistaugliche Lösungen.



Hans-Ulrich Sesselmann
ist IT-Supervisor
bei Deep Silver FISHLABS.

Der IT-Systemkaufmann arbeitet bereits seit 15 Jahren im IT-Bereich und seit 5 Jahren in der Games-Branche. Seit November 2012 war er bei Deep Silver FISHLABS als WebDev tätig, im März 2013 wurde er IT-Supervisor. Zuvor war er unter anderem für Bi-Log, blue cell networks, Upjers und Razer sowie sein eigenes Studio Artoxy tätig. Er ist spezialisiert auf Linux-Server und skalierbare Cloud-Lösungen.



Nils Plaschke
ist Junior Backend Developer
bei Deep Silver FISHLABS.

Im Anschluss an sein duales Studium mit Schwerpunkt Telekommunikationsinformatik bei Alcatel-Lucent Deutschland hat Nils Plaschke im Oktober 2013 seine jetzige Tätigkeit als Junior Backend Developer bei Deep Silver FISHLABS aufgenommen. In dieser Position verantwortet er unter anderem die Weiterentwicklung des Gameservers und der Gamelogik im Java-Backend.

Bei der Entwicklung und Vermarktung unserer 3D-Mobile-Games sind wir lange Zeit der Premium-Positionierung etablierter PC- und Konsolenstudios gefolgt. Die jüngste Vergangenheit hat jedoch gezeigt, dass dieses Geschäftsmodell im App Store und auf Google Play keine Zukunft mehr hat. Somit war auch bei uns Umdenken angesagt. Mit »Galaxy on Fire – Alliances« (GOFA) steht seit knapp einem halben Jahr unser erstes vollwertiges Free2Play-Spiel für iPhone und iPad zum Download bereit (**Abbildung 1**). Der Weg vom Premium-Developer zum F2P-Entwickler war sicherlich der schwerste, den wir in unserer zehnjährigen Unternehmensgeschichte gegangen sind. Und wir sind auch jetzt, gute fünf Monate nach dem eigentlichen Release von »Galaxy on Fire – Alliances«, noch lange nicht am Ziel angekommen.

Seit der branchenweiten Trendwende von »Games as a Product« zu »Games as a Service« bieten auch wir als ehemaliger Premium-Developer unsere Spiele nicht mehr zum Festpreis an, sondern monetarisieren unsere Apps vielmehr über Microtransactions und In-App-Käufe. Anstatt wie in der Vergangenheit ein fertiges und in sich abgeschlossenes

Produkt einmalig zu verkaufen, stellen wir jetzt einen dauerhaften Service bereit, der oftmals erst nach einiger Zeit Früchte trägt und dessen stete Pflege und Erweiterung nicht minder wichtig sind als die ursprüngliche Entwicklung. Denn nur wenn wir immer wieder neuen Content nachliefern und die Performance unserer Titel pausenlos verbessern, werden wir unsere Spieler auf lange Zeit binden und somit besonders gut monetarisieren.

Nachdem wir unser Flaggschiff »Galaxy on Fire 2 HD« im App Store und auf Google Play zuvor bereits rückwirkend auf Paid auf Free2Play umgestellt und an die neue Marktlage angepasst hatten, haben wir mit dem Mobile Strategy MMO »Galaxy on Fire – Alliances« gegen Ende des letzten Jahres unser erstes vollwertiges F2P-Multiplayerspiel an den Mann gebracht. Dies hat selbstredend zu großen Veränderungen in unserem Arbeitsfeld und Zuständigkeitsbereich sowie bei der Teamzusammenstellung geführt. Weil unser aktueller Titel hunderttausenden Spielern auf der ganzen Welt ein gemeinsames, durchgehend zugängliches Spielerlebnis bietet, ist ein sicheres, zuverlässiges und effizient arbeitendes Backend für das erfolgreiche Gelingen des »GOFA«-Projektes unabdingbar. Denn nur mit einer starken Serverarchitektur im Rücken

können wir sicherstellen, dass unsere Spieler jederzeit vollen Zugang zur Spielwelt haben – unabhängig davon, aus welcher Zeitzone oder mit welchem Device sie sich einloggen. Der Service, so lautet die oberste Devise für ein erfolgreiches F2P Multiplayer Game, darf niemals runtergehen. Und genau darin liegt auch die Herausforderung, die den Livebetrieb eines solchen Titels so ungemein spannend macht.

Das Festlegen der Anforderungen

Bevor wir mit der eigentlichen Evaluierung beginnen konnten, haben wir zunächst unsere generellen Anforderungen festgelegt:

- Kommunikation zwischen Mobile Client und Gameserver auf Basis von Protobuf
- Stateless Gameserver
- Relationales Datenmodell (zu Beginn auf postgresSQL basierend)

Daraus ergab sich die folgende, sehr klassisch gehaltene Struktur:

- Ein dynamischer DNS-Dienst zur flexiblen Zuschaltung neuer Loadbalancer auf Basis von HAProxy
- Ein Array von relativ leistungsschwachen Gameservern (skalierbar durch das Hinzufügen neuer Nodes)
- Eine sehr leistungsstarke Master/Slave-replizierte Datenbankmaschine

Als potentiellen Flaschenhals haben wir hier bereits die Datenbank identifiziert, da diese nicht ohne Weiteres skaliert werden kann.

Nachdem die Anforderungen klar waren, wurde ebenso deutlich, dass wir nicht davon ausgehen konnten, dass sich hieran nichts ändern würde. Also haben wir darauf verzichtet, klassische PaaS-Lösungen wie Google App Engine oder Cloud-DB-Anbieter wie Enterprise-DB zu testen, um uns nicht in deren starres Korsett zwängen zu müssen.

Stattdessen haben wir uns entschieden, zunächst diverse Cloudanbieter, namentlich Amazon Webservices, Microsoft Windows Azure und Google Compute Engine, zu evaluieren und mit der Möglichkeit von Colocation Hosting zu vergleichen. Zum damaligen Zeitpunkt waren wir davon überzeugt, dass Colocation schon preislich gesehen unschlagbar sein dürfte, ebenso was die Flexibilität des eigenen Setups angeht.

Weil bereits ausführliche (und nicht besonders ermutigende) Tests zur Leistungsfähigkeit von Amazon EC2 existieren, bei denen besonders die mangelnde Konstanz in der Leistung der gestarteten Maschinen negativ aufgefallen ist, haben wir uns bei der weiteren Evaluierung komplett auf Microsoft und Google konzentriert. Beide Unternehmen waren nach einer ersten Kontaktaufnahme sehr hilfreich, engagiert und haben uns während der Testphase nach Kräften unterstützt. Hierzu zählt das »Whitelisting« der Services ebenso wie ein aktiver Kontakt zu Vertrieb und technischen Mitarbeitern.



Abbildung 1 In einem der größten Kämpfe der Geschichte von »Galaxy on Fire - Alliances« haben die angreifenden Commander insgesamt über 1.500 Carrier und 19 Millionen Truppen in die Schlacht geschickt.

Die Hinzunahme von RightScale

Bereits in den vorbereitenden Gesprächen wurde uns von beiden Parteien die Zusammenarbeit mit RightScale empfohlen, da wir leider nur über begrenzte Admin-Kapazitäten verfügten (und verfügen). RightScale ist ein Cloud Management Tool, mit dem man auf eine Vielzahl vorgefertigter Servertemplates zugreifen kann. Diese Templates enthalten dann startbare OS-Snapshots (wir nutzen CentOS 6.5), drei Kategorien von ausführbaren und variabel konfigurierbaren Scripts (Boottime, Operational, Decomission) sowie Alerts zur Server- und Serviceüberwachung. Alle diese Templates können geklont und individuell angepasst werden. Es ist ebenfalls möglich, ein Template von Grund auf selbst zu designen, wir haben uns aber eher darauf konzentriert, bereits bestehende Lösungen an unseren Bedarf anzupassen.

Weil RightScale in der Lage ist, verschiedene Cloudanbieter zu bedienen und die Firma bei der Evaluierung ebenfalls sehr kooperativ war, konnten wir hier mit minimalem Aufwand unsere gewünschte Infrastruktur aufbauen und dann auf die zu testende Cloud übertragen.

Da wir, wie oben bereits erwähnt, die Datenbank als potenziellen Flaschenhals identifizieren konnten, haben wir uns primär auf die Leistungsfähigkeit der großen Instanztypen konzentriert.

Hierbei handelt es sich um die in Tabelle 1 aufgeführten Konfigurationen.

Anbieter	Typ	vCPUs	RAM	Disk
Google	n1-highmem-8	8	52 GB	100 GB
Microsoft	A7	8	56 GB	100 GB

Tabelle 1 Die Konfigurationen sind heute teilweise nicht mehr verfügbar. Google hat zum Beispiel die Scratchdisks inzwischen abgeschafft.

Wir haben nun zu Testzwecken jeweils eine Maschine gebootet und sie mit pgBench getestet (**Abbildung 2**).

Die Ergebnisse zeigen sehr deutlich, dass Google hält, was es verspricht: konstante Leistung. Während die Microsoft-Maschine zwar etwas performanter als die verglichene Google-Instanz war, bietet Google deutlich weniger Abweichungen. Dagegen zeigten sich bei Microsoft teils starke Schwankungen.

Nach mehr als einem Jahr Arbeit mit Google Compute Engine und insgesamt mehr als 1.000 verwendeten Instanzen (durch Testing, Autoscaling etc.) können wir sagen, dass wir bislang nur zwei Fälle hatten, in denen die Leistung nicht wie angestrebt erbracht wurde – ein beeindruckendes Ergebnis.

Weitere Testkriterien waren Bootzeiten, Datendurchsatz, Leistungsumfang/Tools, Dokumentation und letztlich auch der Preis. Generell lagen beide Clouds hier auf Augenhöhe. Den Ausschlag für die Entscheidung für Google Compute Engine hat dann die Konstanz in der Leistung gegeben sowie das etwas attraktivere Angebot im Pricing.

Der Aufbau der Deployment-Struktur

Parallel zur eigentlichen Cloud-Evaluierung haben wir ebenfalls die Deployment-Struktur mit RightScale aufgebaut und durchautomatisiert. So entstanden angepasste Templates für MySQL-Server in Master/Slave-Replikation, speziell zugeschnittene Templates für unsere Gameserver und Loadbalancer sowie ein extra Template für asynchrone Jobarbeiten wie zum Beispiel Leaderboard-Berechnungen. All diese

Templates beinhalten Scripts für die individuelle Anpassung der einzelnen Maschinen, etwa Gameserver-Konfigurationsdateien oder Ähnliches. Diese Scripts sind so generisch geschrieben, dass sie oftmals in mehreren unterschiedlichen Templates zum Einsatz kommen können.

Der ursprüngliche Entwurf sah vor, das Loadbalancing zwischen schreibenden und lesenden Zugriffen nicht auf Applikationsseite, sondern für die Applikation transparent durch pgPool zu gestalten (**Abbildung 3**). Um auch dieses System flexibel zu halten, kamen wir nach einigen Versuchen, mit pgPool auf eigenen VMs zwischen Datenbanken und Applikationsservern zu installieren, auf die Lösung, auf jedem Gameserver eine eigene pgPool-Instanz lokal zu installieren. Mittels der via RightScale zur Verfügung gestellten Scriptmöglichkeiten war es auch kein Problem, die Konfigurationen jeweils individuell sauber und automatisch zu installieren.

Die Closed Beta mit echten Usern

Auf Basis dieses ersten Setups führten wir nun automatisierte Lasttests aus, um in etwa zu bestimmen, wie viele simultane User (CCU = Concurrent Users) wir ungefähr verkraften können. Wir gingen hierbei davon aus, dass User dann als simultan gelten, wenn sie innerhalb der letzten fünf Minuten eine schreibende Aktion in der Datenbank ausgeführt haben. Die Ergebnisse waren sehr zufriedenstellend und wir befanden unsere Grundlage für tauglich.

Mit diesem grundsoliden Fundament waren wir nun endlich bereit, unser Spiel in die

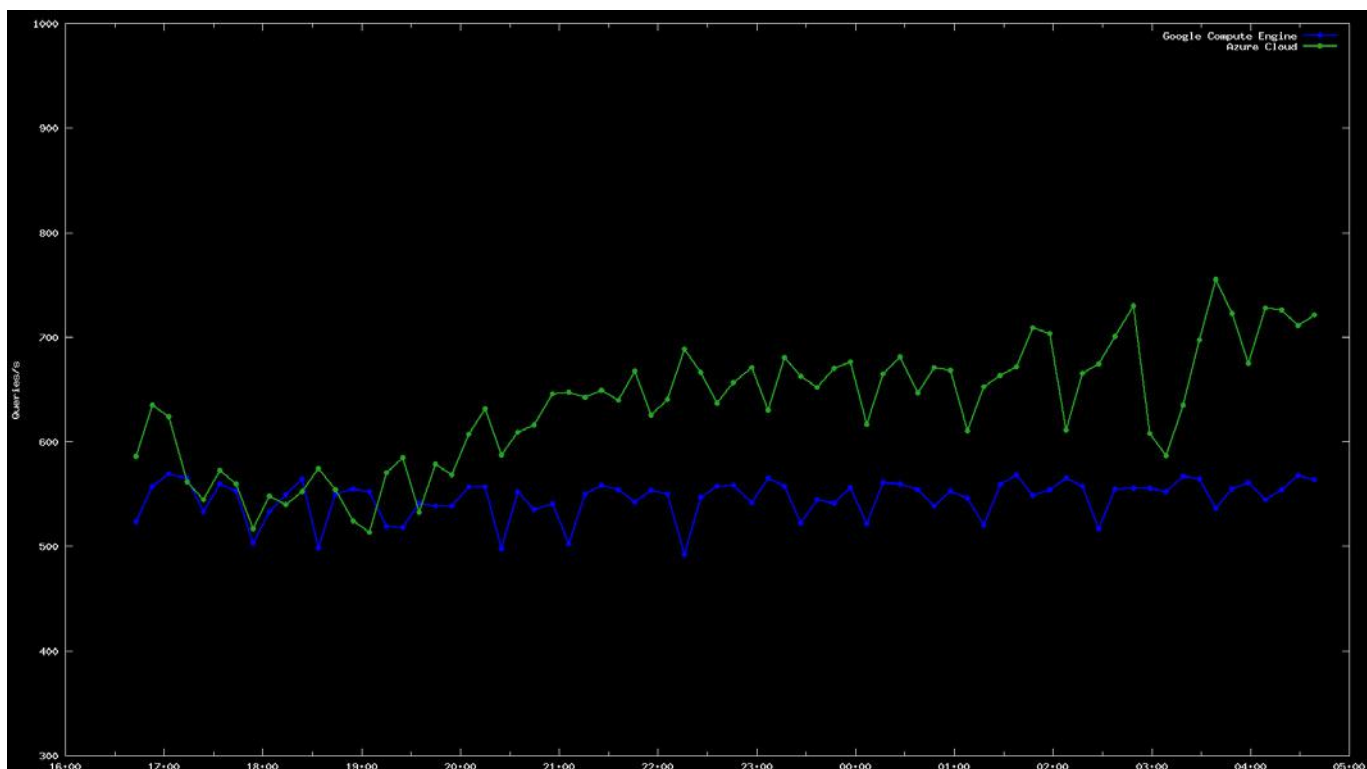


Abbildung 2 Ergebnis der Systemleistungstests zwischen Windows Azure und Google Compute Engine auf Basis von pgBench.

Wildnis des geschlossenen Beta-Tests zu entlassen. Der Plan, echte User auf die Server zu lassen, um nun auch endlich echtes Verhalten sehen zu können, war gut, und wie jeder gute Plan überlebte er nur bis zum ersten Kontakt mit der Realität.

Die User verhielten sich nämlich offenbar komplett anders, als wir uns für unsere Tests vorgestellt hatten. Die Laufzeiten der einzelnen Anfragen nahmen außerdem mit steigender Userzahl exponentiell zu, ebenso die Last auf den Datenbanken. Außerdem war zu beobachten, dass das von uns angestrebte Loadbalancing zur Datenbank mit pgPool in keinsten Weise den gewünschten Effekt brachte. Die Slave-Instanz der Datenbank war beinahe arbeitslos, während die Masterinstanz bereits bei weniger als 100 CCUs Auslastungen von 70 bis 80 Prozent zeigte. Dies war überraschend, weil unsere Analysen ein Verhältnis von ca. 7:1 lesenden zu schreibenden Zugriffen ergeben hatte.

Erste Maßnahme hier war, pgPool abzuschalten, da die Master/Slave-Verteilung augenscheinlich sinnlos war. Allein hierdurch erhielten wir eine ungefähre Beschleunigung der Antwortzeiten seitens der Applikationsserver um den schier unglaublichen Faktor 9.

Die Umstellung von postgresSQL auf MySQL

Da wir anwendungsseitig auf Hibernate als Datenbank-Abstraktionslayer setzen, konnten wir relativ leicht eine Umstellung auf MySQL durchführen und testen. Diese »trockenen« Tests haben bereits einen enormen Geschwindigkeitszuwachs gezeigt (ca. nochmals etwa Faktor 3), woraus wir schlossen, dass entweder unsere postgres-Konfiguration gnadenlos schlecht war oder unsere Anwendung schlicht auf MySQL besser funktioniert (Abbildung 4).

Die Lehre daraus ist keinesfalls pro MySQL oder contra postgresSQL, sondern schlicht und ergreifend, dass wir zukünftig unsere neuen Projekte mit ihrem jeweiligen Datenmodell auf verschiedenen Datenbanken testen und das jeweils geeignetste verwenden werden.

Schlechte Konfiguration hin, bessere Funktion auf MySQL her, diese Ergebnisse rechtfertigten den Aufwand, die bestehenden Daten der BETA-Spielwelt zu konvertieren. Diese Konvertierung erforderte einiges an Vorbereitung und Testing, um sicherzustellen, dass die Daten der User den Weg von postgresSQL zu MySQL unbeschadet überstehen. Mittels Shelltools wie sed und regulären Ausdrücken sowie einer ordentlichen Portion Geduld (hier mussten ca. 6 GB große Datenbank-Dumps verarbeitet werden) war es uns aber dann doch gelungen. Das Ergebnis spricht für sich (Abbildung 5).

Der Preis für diese Umstellung war der Verzicht auf eine für die Applikation transparente Master/Slave-Lastverteilung. Trotzdem waren wir nun in der Lage, bis auf ca. 1.600 CCUs hochzuskalieren. Ab diesem Punkt erwiesen sich die

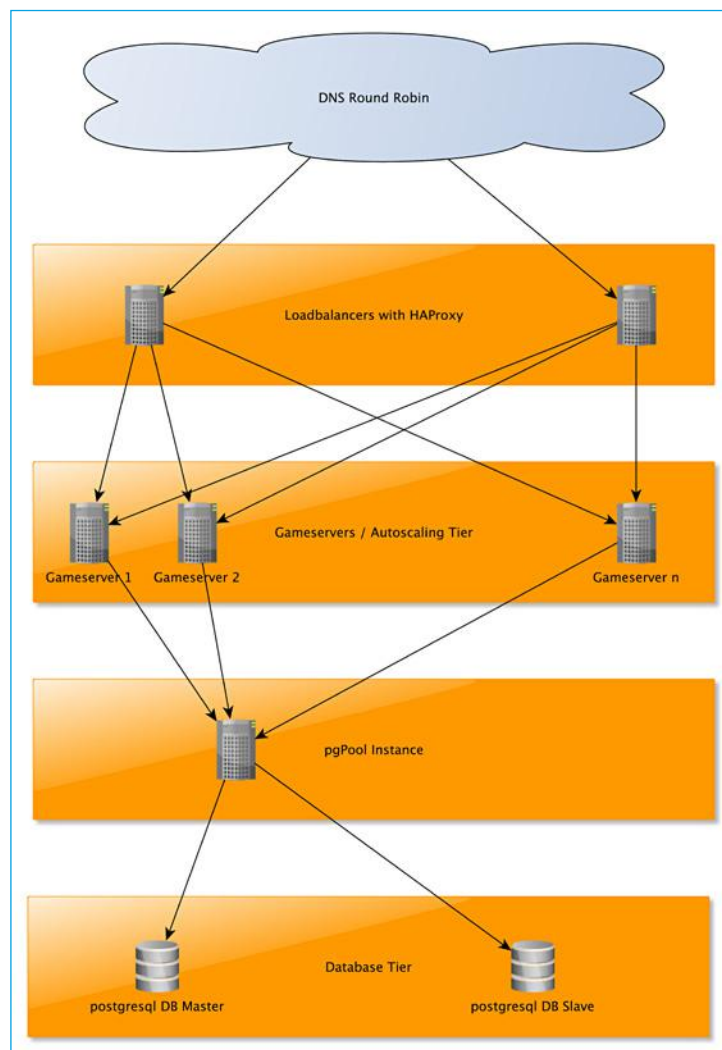


Abbildung 3 Ursprünglich geplanter Aufbau des Server-Setups auf Basis von postgresql und pgPool.

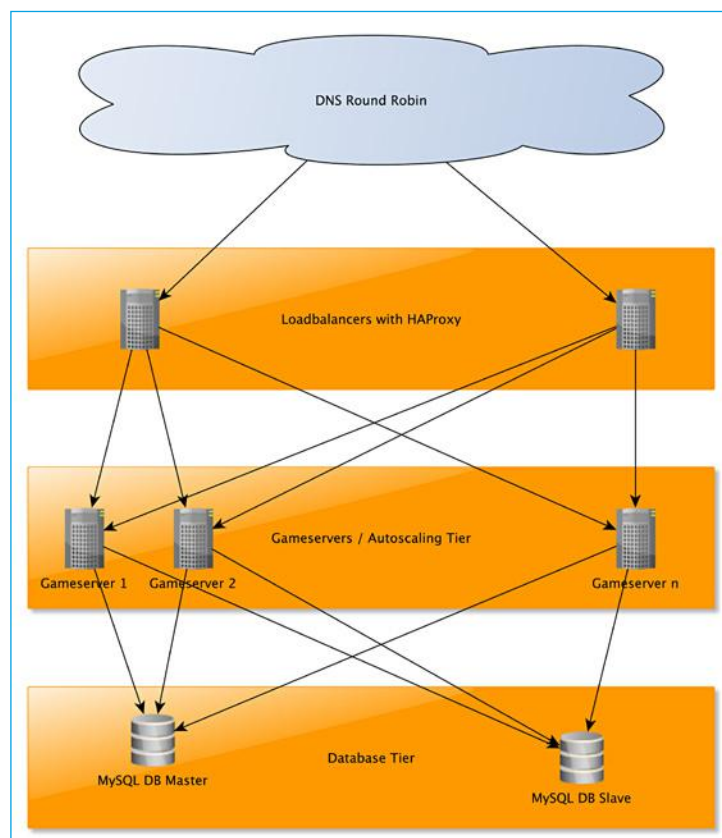


Abbildung 4 Aktueller Aufbau des Server-Setups auf Basis von MySQL.

Festplatten der Datenbank als Flaschenhals. Weil Google zu diesem Zeitpunkt noch keine zufriedenstellende Lösung für das Problem von hohem IOPS-Bedarf anbot, konzentrierten wir uns darauf, die Lastverteilung zwischen schreibenden (Master) und lesenden (slave) Datenbankzugriffen in der Applikation selbst zu realisieren. Dies wurde schlicht durch eine zweite Read-only Connection umgesetzt, und innerhalb der Anwendung entscheidet nun der Entwickler selbst, ob an dieser Stelle schreibende Zugriffe notwendig sind oder ob die Anfrage nur lesend bearbeitet werden kann.

Nach dieser Umstellung zeigte sich auch endlich das von der Master/Slave-Konzeption erwartete Bild: Der Master liegt nun lasttechnisch deutlich unter der Last des Slaves.

Die Anpassung der Plattform an die aktuellen Realitäten

Seither gab es noch genügend Gelegenheiten, um sowohl Plattform als auch Applikation an die aktuellen Realitäten anzupassen. Gerade plattformseitig leistet hier die Cloudlösung unschätzbare Vorteile. Aktuellstes Beispiel war wieder die Festplatte der Datenbank als Flaschenhals. Hier führte ein Engpass dazu,

dass die Replikation nicht mehr schnell genug funktionierte, Master und Slave so asynchron wurden und die Applikation dadurch naturgemäß Probleme bekam. Um das zu beheben, half nur noch, höhere IOPS bereitzustellen. Glücklicherweise hat Google an dieser Stelle seit dem letzten Zwischenfall stark nachgebessert und ist nun in der Lage, auch die IOPS zu skalieren. Je größer die Festplatte, desto höher die IOPS.

Unsere Applikation benötigt bislang Spitzen von bis zu 450 lesenden IOPS, was durch eine genügend große Platte realisiert werden kann. Weil klassische Festplatten lediglich an die 200 IOPS schaffen, hätte man hier bei klassischem Colocation Hosting ein nur teuer zu lösendes Problem. Man benötigt eine intelligente Storage-Lösung, und die kostet bekanntlich viel Geld. SSDs als Alternative scheiden wegen der hohen Aktivität ebenfalls aus, denn bei der aktuellen Last dürften die Flashspeicher nach wenigen Wochen bis Monaten defekt sein.

Man kann hier also sehen, dass ein Live-Backend niemals »fertig« ist, sondern eben exakt das, was der Name sagt: ein scheinbar lebendes Gebilde, dass seine Ansprüche ändert und sich andauernd an neue Gegebenheiten anpassen muss. Ohne eine stabile und gleichzeitig flexible Serverarchitektur ist kein Live-Betrieb möglich, und ohne fähige, motivierte und kreative Mitarbeiter in diesem Bereich gibt es kein solches Backend. Aus diesem Grund werden wir unser Backend-Team in Zukunft noch weiter ausbauen und gewährleisten, dass wir in diesem Bereich besonders stark aufgestellt sind.

Die Definition des Backends als Herz des Spiels

Kommen wir zum Backend, dem unsichtbaren Herzen des Spiels, wo die Logik des Spiels auf die Anfragen der User trifft.

Das Backend ist komplett in Java programmiert, zum einen weil der Gameserver so universell auf unterschiedlicher Hardware zu betreiben ist, zum anderen weil mit Java ein

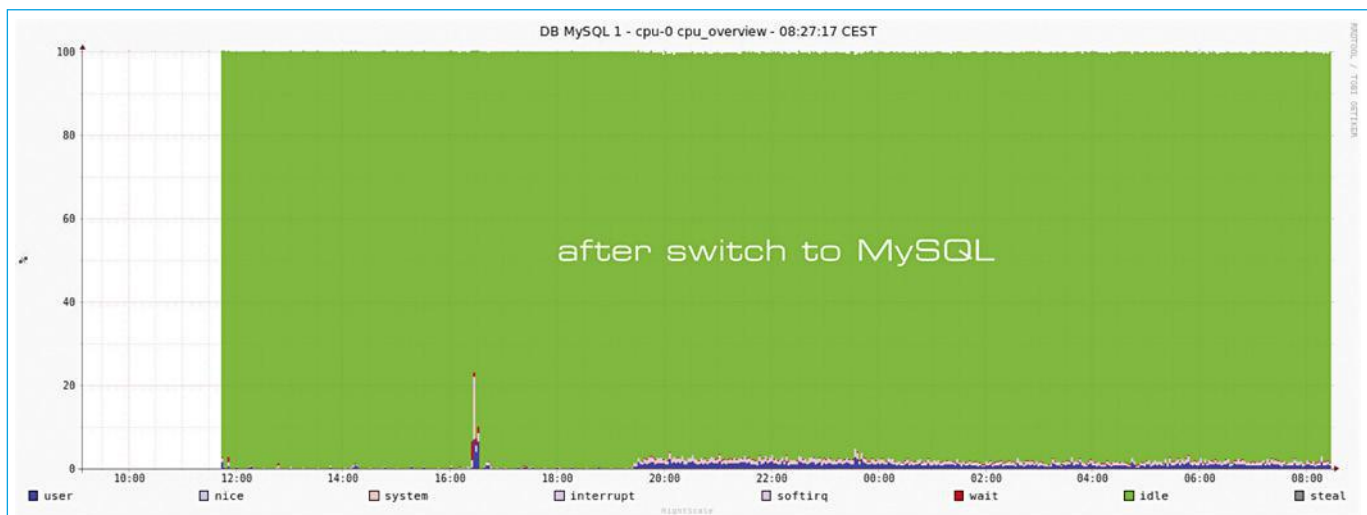
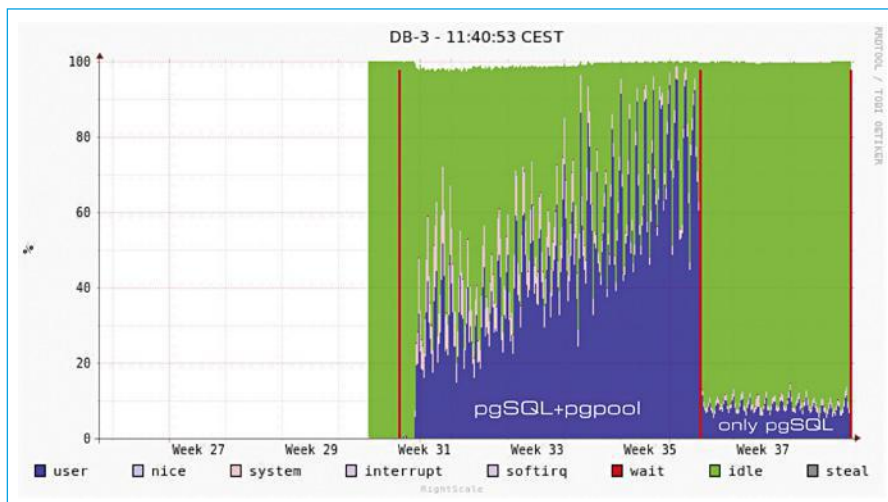


Abbildung 5 Der Umstieg von pgPool (oben) auf MySQL (unten) sorgte für einen deutlichen Rückgang des Workloads.

sehr schneller Iterationszyklus möglich ist. Der ausschlaggebende Grund für Java waren aber vor allem auch die zahlreichen zusätzlichen quelloffenen Libraries die in vielen großen Unternehmen eingesetzt werden und Java somit zu einer der umfangreichsten Plattformen macht.

Während es im Bereich von Java bereits viele fertige Lösungen für ein TCP Backend gibt (VertX, Play, Spring), die allesamt auf großartigen Open Source Libraries aufbauen, haben wir beschlossen, den Gameserver mit seinen Komponenten auf Basis der Libraries selbst zu einem Framework zusammenzubauen.

So ist es zwar möglich, dass die Grundkomponenten gleich sind, wir aber trotzdem die volle Kontrolle über die Applikation und ihr Innenleben haben, die bei einigen Frameworks auf Kosten der Flexibilität und Schnelligkeit verloren geht.

Der Gameserver setzt sich – wie viele Frameworks in diesem Bereich – aus folgenden grundsoliden Komponenten zusammen:

- **Protobuf:** für die effiziente Kommunikation zwischen Client, Lobby und den Gameservern
- **C3PO:** unsere Schnittstelle zur Datenbank mit zwei verschiedenen Pools, einen für den Master und einen für den Slave
- **Hibernate:** unsere Wahl für ORM, ohne viel Programmieraufwand, jedoch mit der Möglichkeit, auch native Lösungen zu entwickeln, sollten die Konventionen nicht unseren Ansprüchen genügen
- **LogBack:** alles, was geloggt werden muss, läuft bei uns über SLF4J und LogBack
- **Quartz Scheduler:** alle nicht proaktiven Jobs werden über den Quartz Scheduler angestoßen und ausgeführt, wobei der Scheduler im Cluster der Gameserver läuft

Generell kann man sagen, dass sich im Gameserver hauptsächlich Open-Source-Lösungen tummeln – ein Zeichen dafür, dass sich die Open-Source-Gemeinde vor hochperformanten Anwendungen nicht verstecken muss.

Die Authentifizierung von Requests mithilfe von User Tokens

Da es sich bei den Gameservern um stateless Server handelt, die somit einfach horizontal skaliert werden können, muss jede Anfrage am Gameserver durch sogenannte Usertokens authentifiziert werden, sodass für jede Anfrage der anzufragende User eindeutig identifiziert und die angefragten Aktionen in seinem Kontext ausgeführt werden können. Der Request-Workflow kann mit dem Diagramm in **Abbildung 7** grob umschrieben werden.

Anhand des Diagramms wird auch die Strategie deutlich, die für die Interaktion mit der Datenbank gewählt wurde:

- Jede Aktion wird mit einer eigenen Connection von dem entsprechenden Datenbank-Pool ausgestattet, sodass innerhalb einer Aktion die Daten immer konsistent sind.

- Da mehrere Anfragen eines Klienten gleichzeitig bearbeitet werden können, ist es notwendig, die Laufzeiten der einzelnen Anfragen so gering wie möglich zu halten, weil die einzelnen Aktionen einer Anfrage interferieren und somit die Integrität der Daten verletzen können.
- Mit dieser Strategie können auch die meisten Aktionen ohne Probleme durchgeführt werden, in Einzelfällen kann es aber – bedingt durch die Größe der zu verarbeitenden Daten oder das Warten auf die Datenbank – zu längeren Laufzeiten kommen.
- Um dieses Problem zu umgehen, werden bei kritischen Aktionen alle Datenbank-Updates am Ende ausgeführt. Die Datenbank wird also nicht über die Laufzeit der Aktion hinweg upgedatet, sondern ein einziges Mal am Ende der Aktion; alle notwendigen Updates werden während der Aktion in Containern gespeichert und am Ende ausgewertet.

Die Implementierung eines zweigleisigen Jobsystems

Ein anderes Problem, das jedes Onlinespiel zu lösen hat, ist die Abarbeitung der von den Spielern gestarteten Jobs – sei es die Fertigstellung einer riesigen Flotte oder der Befehl zum Angriff auf einen Planeten bei einer Multiplayer-Raumschlacht (**Abbildung 8**). Die erste Lösung war ein sogenanntes proaktives Jobsystem, bei dem die Jobs über eine eigene Aktion vom Spieler selbst aufgelöst werden. Bei einem schreibenden Zugriff auf eine Kolonie oder beim Abfragen



Seit Dezember 2013 ist FISHLABS das Mobile-Studio von Koch Media unter dem Publishing-Label Deep Silver und sucht neue Teammitglieder, allen voran einen DevOps Specialist und einen Backend Architect. Ausführliche Informationen zu den beiden offenen Stellen sowie einer Vielzahl weiterer Vakanzen sind auf der offiziellen Website www.dsfishlabs.com einsehbar.

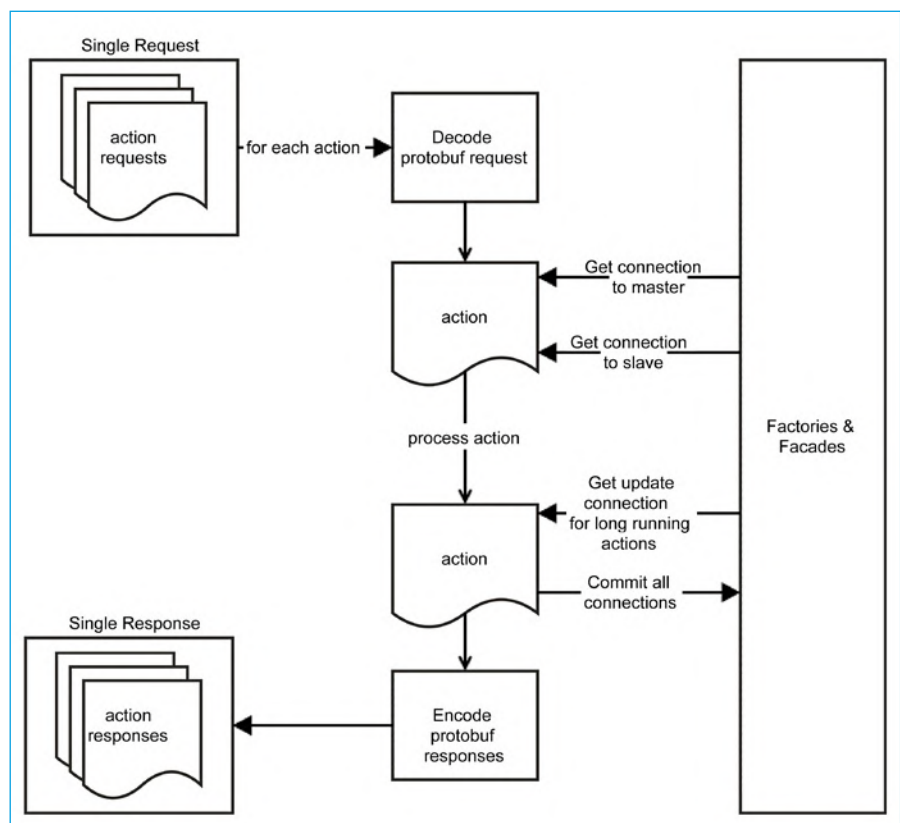


Abbildung 6 Ein grober Überblick über den Request-Workflow im Gameserver.

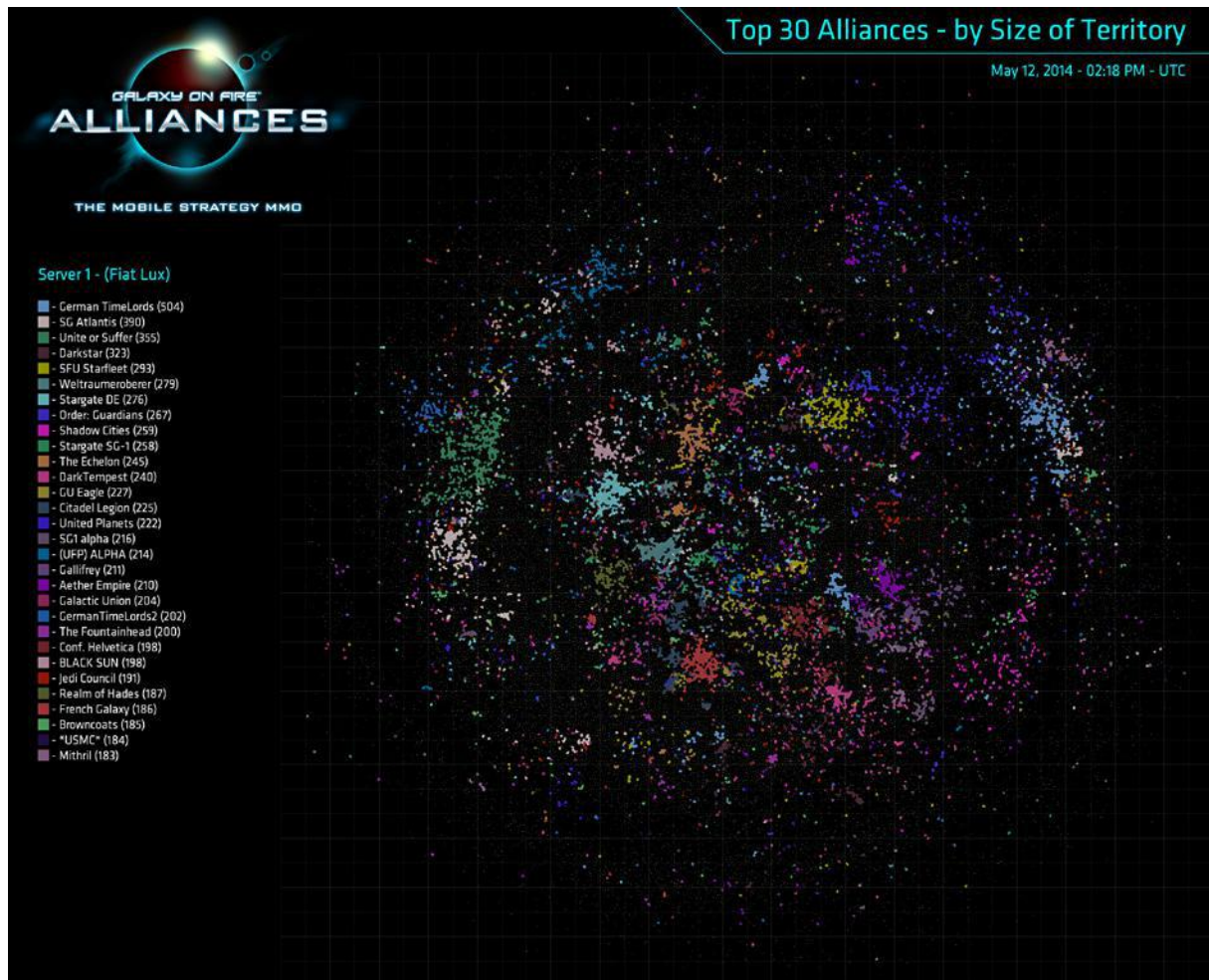


Abbildung 7 Die komplette Sternenkarte von Server 1 mit Fokus auf die Hoheitsgebiete der Top-30-Allianzen.

der Flotten wurden alle Jobs aufgelöst, die zu diesem Zeitpunkt bereits beendet waren.

Dies führte jedoch bei größeren Kämpfen oder Spielern, die über längere Zeiträume inaktiv waren, immer wieder zu Problemen, weil die benötigte Zeit für die entsprechende Joblösung sehr groß war. Aus diesem Grund wurde neben dem proaktiven Jobsystem ein zweites System auf Basis des Quartz Schedulers aufgebaut, das nun alle Jobs, die sofort aufgelöst werden müssen, auflösen kann.

Das interessante am Quartz Scheduler ist die Tatsache, dass dieser auch in einem Clustered Environment laufen kann, also jeder Game-server in der Lage ist, die anfallenden Jobs abzuarbeiten. Somit ist gewährleistet, dass alle Nodes (Gameserver) zu gleichen Teilen mit den Anfragen ausgelastet werden können, ohne einen speziellen Server für die Auflösung bereitstellen zu müssen. Eine weitere interessante Tatsache ist die Failover-Funktion des Quartz Schedulers. Damit ist die Redundanz auch auf der Jobseite gewährleistet.

Die Entscheidung für den Quartz Scheduler

Es wurde jedoch nicht sofort die Quartz Scheduler Library in Betracht gezogen. Die Evaluie-

rung für das zweite Jobsystem umfasste unter anderem auch die Kandidaten beanstalkd, RabbitMQ und IronMQ.

Im Gegensatz zu Quartz bieten diese jedoch ein komplettes Messaging System an, sodass alle anderen, bis auf beanstalkd, schon früh ignoriert werden konnten. Gegen beanstalkd sprach die Tatsache, dass es nicht ohne weiteren Aufwand in die bestehende Applikation eingebettet werden konnte. Zwar gibt es Client Libraries, die man in Java ohne Probleme einbinden kann, allerdings benötigt man neben den Clients noch einen Server, der die Jobs entgegennimmt. Quartz hingegen speichert die Jobs in einer relationalen Datenbank und passte somit hervorragend in die bestehende Struktur, ohne großartige Änderungen vornehmen zu müssen. Zudem profitiert auch der Heartbeat von Quartz, weil so aufwendige Tasks wesentlich einfacher und eleganter realisiert werden konnten.

Das neue Jobsystem befindet sich im Moment in der Testphase, verspricht aber bereits jetzt schon eine bessere Parallelisierung der Jobs, eine losere Kopplung der einzelnen Komponenten und eine flexiblere Anpassung an weiteren Content und Funktionen, was für ein F2P-Spiel unablässig ist.

Die Berechnung von Leaderboards in Echtzeit

Ein weiterer wichtiger Bestandteil jedes Onlinespiels sind Leaderboards ([Abbildung 9](#)), Events und Achievements, die in regelmäßigen Abständen berechnet werden müssen. Wir haben eine Funktion integriert, die es erlaubt, den Server im Heartbeat-Modus zu starten, der alle periodisch anfallenden Aufgaben übernimmt. Zuerst basierend auf der javainternen TimerClass und ähnlichen Klassen haben wir nach und nach diesen Heartbeat auch auf die Quartz Scheduler Library umgestellt, die genau für diese Zwecke entwickelt wurde. Generell kann man sagen, dass gerade hier eine frühere Umstellung viele Probleme verhindert hätte, aber wie bei jedem neuen Projekt lernt man immer etwas Neues hinzu und zieht Lehren aus den begangenen Fehlern.

Die Nutzung von Lambdas und Streams in Java8

Die letzte größere Neuerung, die kürzlich in das Backend integriert wurde (sich allerdings auch noch im Testmodus befindet), ist Java8. Mit Java8 halten Lambdas und unser neues Lieblingskonstrukt – die Streams – Einzug in die Java-Welt. Lambdas machen unseren Code leserlicher, einfacher und wartungsfreundlicher. Wer aus dem Bereich von JavaScript kommt, der weiß, wovon wir reden.

Doch gerade die Streams haben sich in unserer Applikation gut integriert: Wurden früher alle Collections mittels einer foreach-Schleife durchlaufen, die die JVM zu einer seriellen Abarbeitung zwingen, können nun gerade im Bereich des BattleModuls viele Berechnungen auf mehrere Threads und somit mehrere Kerne ausgelagert werden. Das war vorher nur mit viel Aufwand möglich, der sich zudem in den meisten Fällen nicht gelohnt hat. Streams und Lambdas sorgen aber auch dafür, dass vorher sehr verschachtelter und langer Code nun übersichtlich und kurz bleibt – und tragen letzten Endes so zu einem sauberen Programmierstil bei.

Die Berücksichtigung zukünftiger Probleme und Herausforderungen

Selbst wenn am Gameserver bereits viel umgebaut und beschlossen wurde, sind auch hier immer wieder Verbesserungen gewünscht und nötig. Sollte in Zukunft die Last auf den Slave weiter steigen, was sehr wahrscheinlich ist, dann besteht die Möglichkeit, mehrere Slaves aufzustellen und per Loadbalancer zu schalten, wobei der Gameserver nur die Loadbalancer anspricht und somit nur einen Slave sieht. Wer Spaß an der Entwicklung von Client-Server-Anwendungen hat und auch in Java kein totes Pferd sieht, der wird hier in der Position des Backend-Developers sehr glücklich.

Neue Entwicklungen auszuprobieren so wie alte Konstrukte zu hinterfragen, ist ein



Abbildung 8 Eine Multiplayer-Raumschlacht mit Dutzenden von Angreifern und Hunderttausenden von Schiffen.



Abbildung 9 In »Galaxy on Fire - Alliances« gibt es diverse Leaderboards für die erfolgreichsten Einzelspieler und Allianzen. Diese werden mehrmals täglich aktualisiert und bilden stets die aktuellen Kräfteverhältnisse ab.

integraler Bestandteil der zu leistenden Arbeit. Nicht selten kommt es vor, dass ganze Klassen umgeschrieben werden, um noch das letzte bisschen Performance aus einer speziellen Aktion herauszupressen.

Trotzdem ist es notwendig, auch die weitere Pflege und Erweiterbarkeit des Servers im Auge zu behalten. Wer hier ein Gespür für Verbesserungen hat, kann dafür sorgen, dass sich alle Spieler gleichzeitig und ohne Probleme in großen Weltraumschlachten messen können.

Nils Plaschke, Hans-Ulrich Sesselmann

VON FACEBOOK ZU MOBILE

Kabam war eines der ersten Unternehmen, die den Sprung von Facebook auf Mobile geschafft haben. Andrew Sheppard über das richtige Timing, organisches Wachstum und die Wahl durch die Kunden.



Andrew Sheppard
ist Präsident
von Kabam Studios.

Andrew hat einen Masterabschluss in Business Administration von der UC Berkeley's Haas School of Business sowie einen Bachelor in Wirtschaftswissenschaften vom Pomona College. Er hat die Online- und Free2Play-Strategie-Abteilung bei Electronic Arts geleitet. Außerdem war er daran beteiligt, dass sich hi5 von einem sozialen Netzwerk zu einer Social Gaming Plattform entwickelt hat. Seit 2009 arbeitet er bei Kabam, unter seiner Leitung sind in der Zwischenzeit sieben Spiele entstanden. Zudem verwaltet er die vier Entwicklerstudios von Kabam in San Francisco, Peking, Vancouver und Austin, Texas.

Making Games Wann und warum habt ihr euch zur Arbeit an Spielen für Mobile entschieden? Wie sahen eure ersten Schritte aus?

Andrew Sheppard Ich erinnere mich noch sehr gut an eine Unterhaltung, die ich 2011 mit unserem CEO Kevin Chou hatte. Er ist fantastisch darin, die Entwicklung der Spieleindustrie vorherzusehen und so den Trends vorzugreifen. Und schon damals sah Kevin die Möglichkeit – oder vielmehr die Notwendigkeit – zur Entwicklung für Mobile. Gemeinsam haben wir dann ein Rahmenkonzept entwickelt. Dazu gehörte, unser Studio in Peking komplett auf unsere ersten Mobile-Titel zu konzentrieren. Das waren damals Ports von bereits existierenden Spielen: »Kingdoms of Camelot« und »Dragons of Atlantis«. Seitdem hab ich mich darauf konzentriert, unsere Mobile-Ressourcen im gesamten Unternehmen aufzubauen und zu erweitern.

Das Studio in Peking hat das Spektrum seiner Möglichkeiten über Strategiespiele hinaus weiterentwickelt. Jetzt entstehen dort auch Battle Card Games und ein Actionspiel namens »Glass Zone«, das noch dieses Jahr erscheinen wird. Unser Studio in Nordamerika hat den Franchise-Titel »Fast & Furious 6: The Game« auf den Markt gebracht, eines der am schnellsten wachsenden Mobile-Spiele mit über 33 Millionen Downloads. Vergangenen Dezember erschien außerdem das Spiel

»Dark District« für iOS-Geräte. Es ist das erste 3D-Tower-Defense-Spiel mit direkter Kontrolle, daher waren wir sehr aufgeregt, als wir den Titel veröffentlicht haben.

Generell haben wir folgende Herangehensweise gewählt: Ein Team fokussiert sich auf die Lösung eines Problems und im Anschluss wird auf dem Wissen dieses Kompetenzzentrums aufgebaut – zuerst lokal, später global.

Ich glaube, dass darin das Geheimnis unseres Erfolgs mit Mobile-Spielen liegt. Wir waren damals das erste Unternehmen, das sich vom Social-Gaming-Bereich gelöst hat. Meiner Meinung nach sind wir aus diesem Grund in der Lage, sowohl den Umfang unseres Geschäfts als auch den Umfang unserer Entwicklungskapazitäten optimal zu nutzen.

Der Publishing-Zweig unseres Unternehmens ist erst etwa ein Jahr alt. Unser Team hat großartige Arbeit geleistet und Kontakte zu den besten Spieleentwicklern der Welt hergestellt. Noch in diesem Jahr werden wir eine wichtige Rolle dabei spielen, einige der besten asiatischen Spiele auf die westlichen Märkte zu bringen.

Making Games Kabam ist in den vergangenen Jahren sehr schnell gewachsen. Wie lässt sich so etwas managen?

Andrew Sheppard Es war gleichzeitig schwieriger und einfacher, als ich es mir hätte vorstellen können. Als ich zu Kabam kam, haben dort 30 Leute gearbeitet. Jetzt sind wir bei rund 700. Aus heutiger Sicht ist es schwierig zu sagen, welche Taktik genau wir angewendet haben. Das frühe Wachstum war durch Neueinstellungen vollkommen organisch und hat sich auf San Francisco konzentriert. Heute haben wir Niederlassungen in Peking, Seoul, Vancouver, San Francisco,

Austin, Berlin, London und Luxemburg. Durch diese dezentrale Struktur sind wir in der Lage, schneller zu wachsen.

Unsere Entwicklungszentren sind in Peking, Vancouver und San Francisco. An allen anderen Standorten mit Ausnahme von Luxemburg haben wir Sales- und Marketing-Büros. Natürlich gab es im Lauf der Zeit auch einige Übernahmen. Ich erinnere mich noch sehr

»Ein Team hat die Lösung eines Problems fokussiert, im Anschluss haben wir auf dem Wissen dieses Kompetenzzentrums aufgebaut.«



»Dragons of Atlantis« (oben) und »Kingdoms of Camelot« (unten) waren die ersten Mobile Games von Kabam. Die Portierung der Internetspiele übernahm das Studio in Peking.

gut an den Kauf von WonderHill. Zusammen haben wir »Dragons of Atlantis« zu einem unserer größten Franchises gemacht. Wir haben außerdem das Studio Exploding Barrel Games in Vancouver übernommen, das dann zu Kabam Vancouver wurde. Dieses Team hat »Fast & Furious 6: The Game« entwickelt und bringt AAA-Erfahrung mit. Die wird für viele unserer zukünftigen Spiele von zentraler Bedeutung



»Facebook ist unglaublich wichtig für uns.«



»Dark District« ist im Dezember 2013 erschienen: »Es ist das erste 3D-Tower-Defense-Spiel mit direkter Kontrolle.«

sein. Doch beide Übernahmen waren verhältnismäßig klein, insgesamt waren es nicht mehr als 75 bis 100 Mitarbeiter.

Making Games Welche Bedeutung hat Facebook heute noch für eure Spiele?

Andrew Sheppard Facebook ist unglaublich wichtig für uns, da wir schon sehr lange zusammenarbeiten. 30 Prozent unseres Unternehmenswerts kommen aus dem Internet. Einen Großteil davon erwirtschaften wir über unsere eigene Website, der Rest kommt allerdings von Facebook; das ist also ein großer Teil unseres Geschäfts. Ganz zu schweigen davon, dass Facebook mit dem mobilen Ökosystem phänomenale Arbeit geleistet hat. Sie sind unser wichtigster Vertriebspartner.

Making Games Wie viele Personen arbeiten in euren Studios an Facebook-Spielen und wie viele arbeiten an Mobile-Titeln?

Andrew Sheppard Rund 100 Leute sind mit unseren Web-Produkten beschäftigt – und das sind nur die Leute, die direkt an den Spielen arbeiten. Das bedeutet, dass 150 weitere Mitarbeiter in diesen Studios mit Marketing beschäftigt sind oder mit dem, was wir »Odd Operations« nennen. Das ist unter anderem die regionale Vermarktung der Spiele oder die direkte Entwicklung.

Making Games Wie unterscheidet ihr bei der Entwicklung zwischen den Spielern im Internet und den Mobile-Usern?

Andrew Sheppard Wir unterscheiden diese Gruppen nicht direkt, versuchen jedoch auch nicht, sie in einen Topf zu werfen. Unser Ansatz bei der Ent-



»Fast & Furious 6: The Game« wurde von Kabam Vancouver (früher Exploding Barrel Games) entwickelt: »Das Team bringt die AAA-Erfahrung mit, die für viele unserer zukünftigen Spiele von zentraler Bedeutung sein wird.«

wicklung eines Produkts – und beim Service für die Spiele – ist die Konzentration auf die Kunden und darauf, was sie uns sagen wollen. Anders gesagt: Als Unternehmen sind wir sehr darauf bedacht, die Kunst und Wissenschaft der Spieleentwicklung zu verbinden. Wir glauben, dass dieser Punkt essenziell ist für die Entwicklung von Free2Play-Spielen für Mobile. Die künstlerische Seite ist es also, fesselnde Spiele zu entwickeln. Sie müssen sich von der Konkurrenz abheben und die Nutzer dazu bringen, dass sie den Titel unbedingt spielen wollen.

Der wissenschaftliche Teil unserer Spieleentwicklung ist meiner Meinung nach einzigartig. Dieser Ansatz ist am besten für bereits laufende Produkte geeignet, weil man hier Echtzeitdaten der User bekommt. Wir wollen verstehen, was die Kunden möchten und was eine Überraschung oder Innovation für sie wäre. Also nutzen wir qualitative und quantitative Forschung – und dann liefern wir diesen Ergebnissen entsprechend. Das ist einer der Gründe, warum wir Free2Play für die demokratischste Art der Spieleentwicklung halten. Am wirkungsvollsten sind die Spielthemen, die auf die Wahl der Spieler hören und sich entsprechend anpassen. Und die Spieler stimmen jeden Tag ab – entweder spielen sie dein Spiel oder sie lassen es bleiben. Unser Ansatz ist also stark auf den Kunden fokussiert.

Making Games Ihr habt folglich keine vorgegebenen Schritte, wenn ihr Spiele für Web oder

Mobile entwickelt, und richtet euch lediglich nach dem Votum der Kunden?

Andrew Sheppard Genau. Wir testen Konzepte mit Fokusgruppen und Spielsessions. Und dann sehen wir uns die gesammelten Hintergrunddaten an und werten aus, ob ein neues Feature oder neue Inhalte so funktionieren, wie wir es erwartet haben.

Making Games Andere Entwickler verfolgen einen Cross-Platform-Ansatz. Warum habt ihr euch dagegen entschieden?

Andrew Sheppard Ein Grund dafür ist die Vielschichtigkeit unserer Spiele. Außerdem wollen wir uns auf den globalen Ausbau konzentrieren und nicht auf die Verbindung unterschiedlicher Plattformen. King ist ein Pionier des wirklich spannenden Cross-Platform-Konzepts. Es war eine kluge Entscheidung, sich so früh damit auseinanderzusetzen – denn in Zukunft wird dieses Thema eine wichtige Rolle spielen. Allerdings ist es wirklich schwierig, vielschichtige Spiele in diesem Konzept umzusetzen. Dieses nuancierte Gameplay findet man in »The Hobbit: Kingdoms of Middle-Earth«, aber auch in Spielen wie »Fast & Furious 6«, die mehr in die Richtung Casual Games gehen und daher insgesamt einfacher zu spielen sind.

Als Unternehmen sind wir stolz darauf, dass wir mit den Entscheidungen in unseren Spielen auch eine soziale Komponente und Konsequenzen verbinden, was King und andere Entwickler in dieser Form nicht betreiben. Das ist der Grund, warum wir es noch nicht angestrebt haben, unsere Spiele über unterschiedliche Plattformen hinweg zu verknüpfen. Es würde mich allerdings nicht überraschen, wenn wir in naher Zukunft einige unserer Tablet-Spiele auch für den PC

veröffentlichen würden. Dafür gibt es allerdings noch keine konkreten Pläne.

»Wir sind sehr darauf bedacht, die Kunst und Wissenschaft der Spieleentwicklung zu verbinden.«

Making Games Mit »The Hobbit« und »Fast & Furious« habt ihr Spiele, die auf einer bereits bestehenden IP aufbauen. Wie beeinflussen diese IPs die Spiele und Spieler?

Andrew Sheppard Jedes unserer laufenden Spiele erreicht irgendwann einen Scheitelpunkt, wenn es um die Attraktivität für den Massenmarkt geht. »Kingdoms of Camelot«, »Dragons of Atlantis«, »The Hobbit«, »Fast & Furious 6« – diese Spiele haben alle eine sehr große Reichweite. Der Vorteil an Lizenzspielen ist, dass man damit User erreicht, die eher von Casual Games angesprochen werden. Wir haben zum Beispiel bei »The Hobbit« User, die sich vorher noch nie an einem Mobile-Titel oder einem Strategiespiel versucht haben. Durch die Suche nach etwas, das zu der Marke gehört, haben sie demnach also etwas gefunden, das für sie neu und aufregend ist und ihnen Spaß macht. Die IP

vergrößert die Reichweite der Spiele. Aber im Endeffekt sorgt nur ein gutes Spiel dafür, dass die User bei der Stange bleiben und hoffentlich auch Geld investieren. Kunst und Wissenschaft sind unsere Ausgangspunkte, aber es schadet keinesfalls, wenn man mit Marken wie »Der Hobbit« und »Fast & Furious« arbeiten kann. Das ist schon ein Segen.

Making Games Was sind die Hauptkriterien, um mit Mobile-Spielen bei Core Gamern erfolgreich zu sein?

Andrew Sheppard Bei diesem Thema kommt es in gewisser Weise auf Definitionen an. Wenn man sich die Basis unserer User ansieht und sich dabei nur auf die Kunden im Mobile-Free2Play-Bereich konzentriert, dann spielen 60 Prozent unserer Spieler auch auf Konsolen. Das ist manchmal ziemlich überraschend, denn wenn man Konsolenspieler fragt, ob sie auch Free2Play-Titel spielen, sagen sie oft »Nein«. Das ist ähnlich, wenn man Spieler von Casual Games fragt, ob sie Videospiele spielen – sie verneinen das genauso oft.

Wir halten es für einen unvermeidlichen Trend, dass die Kunden auf Mobile-Geräte und verstärkt Tablets umsteigen. Gerade der Tablet-Sektor wird in den nächsten fünf Jahren um 42 Prozent wachsen. Aber die User steigen auch auf Free2Play um. Hier ist der Wechsel allerdings schleichender, die Leute spielen nicht schlagartig nur noch Free2Play-Titel. Aber sie spielen definitiv mehr Free2Play-Spiele, und die monatliche Anzahl der Stunden, die sie damit verbringen, wächst stetig. Die Auffassung wird also langsam verschwinden, dass ein Hardcore-Spieler nur Retail-Titel spielt.

Einer meiner zentralen Standpunkte ist, dass es drei unterschiedliche Evolutionsstufen der Spieleindustrie gibt. Die erste Stufe war die Zeit der Arcade-Automaten. Damals waren die Spiele auf die spezielle Hardware ausgelegt und auch darauf, dass sie über Münzeinwurf abgerufen werden konnten. Die zweite Stufe gehörte Konsolen und Retail. In dieser Phase sind Firmen wie EA und Activision aufgestiegen. Die dritte Stufe ist das Free2Play- und Mobile-Phänomen, das die Welt momentan überrollt.

Obwohl stufenübergreifend eine Menge der gleichen Personen an den Spielen gearbeitet haben, waren es immer größtenteils andere Firmen, die in den unterschiedlichen Geschäftsmodellen besonders erfolgreich waren. Ein Unternehmen, das großen Erfolg mit Arcade-Spielen hatte, war beispielsweise nicht für Erfolg in der Konsolen- und Retail-Phase aufgestellt. Ein Teil der Debatte darüber, ob die Kunden Free2Play wollen oder nicht, hängt also auch damit zusammen, dass sich die Leute an dem festhalten, was sie kennen.

Die harte Wahrheit ist, dass Core Gamer genauso Free2Play-Titel spielen und auch immer mehr Zeit damit verbringen.

Making Games Was denkst du, wie sich die Spieleindustrie in den nächsten fünf Jahren entwickeln wird?

Andrew Sheppard Videospiele werden massenmarkttauglicher sein als jemals zuvor. Das Gameplay wird sich aus kürzeren Sessions zusammensetzen und aus Mobile-Free2Play-Sicht mehr auf Core Gamer ausgelegt sein. Aber trotzdem werden die Spiele zugänglich bleiben, eine Eigenschaft, die Konsolentitel im Lauf der Jahre verloren haben. Wenn man sich heute die

neueren Konsolenspiele ansieht, dann würde sich ein Casualspieler damit ziemlich schwer tun.

Ich denke, dass wir auch den Aufstieg einer wirklich globalen IP sehen werden – ein Spiel, das weltweit auf jedem Markt funktioniert, egal ob Japan, Großbritannien, Deutschland, Frankreich oder die USA. Und ich denke, dass es ein Cross-Platform-Titel sein wird. Man sieht

diese Ansätze schon jetzt mit Smartphones und Tablets. Auf beiden Plattformen verhalten sich die Spieler unterschiedlich. Bei den Tablet-Spielern kann man beobachten, dass sie mehr Spiele spielen, länger dabeibleiben und mehr Geld ausgeben. Ein Teil der Spieler wird wieder zurück auf den PC oder sogar die Konsolen abwandern. Ich denke, es wird ein sehr offener und breit aufgestellter globaler Markt sein, der größer ist, als es sich irgendjemand heute auch nur ansatzweise vorstellen kann. Diesen Gedanken finde ich zwar angsteinflößend, aber gleichzeitig auch aufregend.

Free2Play wird meiner Meinung nach in Zukunft das vorherrschende Geschäftsmodell werden. Es gibt in einem Marktplatz mit 300.000 Apps im App Store in Deutschland oder einer Million in den USA einfach so viel Auswahl. Und die Spiele werden insgesamt immer besser. Das Konzept, vom Spieler vorher Geld für einen Titel zu verlangen, der ihm am Ende womöglich nicht mal Spaß macht, erscheint einfach zunehmend verrückt. Man kann die gleiche Entwicklung auch in anderen Medienbereichen beobachten: Musik, Filme, Fernsehen. Inhalte werden immer öfter kostenfrei angeboten und es wird indirekt Geld damit verdient. Ich denke, es wäre engstirnig davon auszugehen, dass sich unsere Sparte anders entwickelt.

Interview: Patricia Geiger

»Wir werden den Aufstieg einer globalen IP sehen – ein Spiel, das weltweit auf jedem Markt funktioniert.«



Spiele wie »The Hobbit: Kingdoms of Middle-Earth« ziehen User an, die sich vorher noch nie an einem Mobile-Titel oder einem Strategiespiel versucht haben.

EINE EIGENE ENGINE BAUEN – TEIL 3

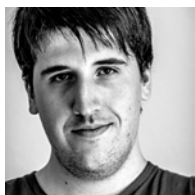
VISUELLE EFFEKTE FÜR DIE NÄCHSTE GENERATION

Deck13 hat für Lords of the Fallen die Multiplattform-Technologie namens Fledge entwickelt. Im dritten Teil der Artikelserie dreht sich alles um die Weiterentwicklung des Renderings und der visuellen Effekte.



David Reinig
ist Engine/Graphics Programmier
bei Deck13 Interactive.

David arbeitet seit Juli 2011 bei Deck13. Aktuell ist er an der Portierung und Optimierung der Xbox-One-Version von Fledge beteiligt. Neben der Pflege und Optimierung vorhandener Systeme der Engine war David maßgeblich für die Portierung der Engine zu DirectX 11 verantwortlich und steuerte einige Subsysteme zur Engine bei, wie zum Beispiel das Partikelsystem und diverse grafische Features.



Benjamin Glatzel
ist Engine/Graphics Programmier
bei Deck13 Interactive.

Benjamin arbeitet seit Anfang 2012 bei Deck13 und ist aktuell für die Portierung und Optimierung der PS4/GNM-Version von Fledge sowie für sämtliche physikalischen Effekte verantwortlich. Neben vielen Low- bis High-Level-Optimierungen hat er zudem zentrale Subsysteme der Engine beigesteuert, wie beispielsweise das auf Software-Rasterisierung basierende Occlusion Culling System, das volumetrische Beleuchtungssystem sowie das Character-IK-System.

Mit der Entscheidung, dass Lords of the Fallen ein Titel für Next-Gen-Konsolen wird, wurde uns schnell klar: Neben dem Grundgerüst von Fledge mussten auch die visuellen Effekte auf

die nächste Ebene gehoben werden. Daher mussten viele Teilbereiche des Renderings entweder grundlegend überholt und erweitert oder komplett neu entwickelt werden.

Der dritte Artikel unserer Serie geht näher auf eine ausgewählte Menge von grafischen Effekten ein, die für Lords of the Fallen entwickelt wurden. Diese Effekte bilden quasi das Konfetti, das die virtuelle Welt für den Spieler erst interessant macht. Neben CPU/GPU-basierten Partikeleffekten unterstützt unsere Engine weitere, seltener genutzte Techniken. Zum Beispiel ein System, um Lichtstreuung in Participating Media (Medien wie Luft oder Wasser, die beim Lichttransport beteiligt sind) in Kombination mit einer Vielzahl von Lichtquellen zu simulieren. Außerdem besitzt Fledge eine vollständige Integration von PhysX und der darauf basierenden Middleware APEX. APEX wird für die physikbasierte, GPU-gestützte Simulation von Kleidung, Parti-

keln sowie dynamisch zerstörbaren Objekten eingesetzt, während PhysX für die Simulation von einfachen Rigid Bodies und komplexeren Ragdolls verwendet wird.

Im letzten Abschnitt dieses Artikels gewähren wir außerdem noch einen Einblick in einige prototypische Systeme, die während der Entwicklung von Lords of the Fallen entstanden sind, jedoch mit hoher Wahrscheinlichkeit – aufgrund des fortgeschrittenen Zustands des Projekts – nicht den Weg in das finale Produkt finden werden. In diesem Kontext gehen wir sowohl auf unser Screen-Space-basiertes Schattensystem als auch unsere Compute-Shader-gestützte Haar-, Fell- und Grassimulation näher ein.

Entwicklung eines volumetrischen Beleuchtungssystems für viele Lichter

Das Beleuchtungssystem zählt zu den zentralen Bestandteilen jeder Game Engine und trägt maßgeblich dazu bei, dass eine in Echtzeit gerenderte 3D-Szene von uns als glaubhaft empfunden werden kann. Für Lords of the Fallen wollten wir unser existierendes, recht einfaches und gewöhnliches System erweitern, sodass wir volumetrische Beleuchtungseffekte für unterschiedlichste Situationen mit einer Vielzahl von



Lichtquellen variierenden Typs simulieren können. Aber was genau ist unter volumetrischen Beleuchtungseffekten zu verstehen?

Für die Simulation einfacher Beleuchtung im Kontext von Lichtquellen wird die Strahldichte (engl. Radiance) ausschließlich auf den Oberflächen der Objekte der 3D-Szene mithilfe des in der Engine vorherrschenden Beleuchtungsmodells und einer Technik zur Berechnung des Schattenwurfs evaluiert. Richten wir also ein Spotlicht auf ein Objekt, wird lediglich das bestrahlte Objekt aufgehellt und eventuelle Objekte in der Umgebung durch dessen Schattenvolumen abgedunkelt. Vergleicht man dies jedoch mit verschiedenen Szenarien aus dem Alltag, wird einem schnell klar, dass dies nur in einer sehr begrenzten Menge von Situationen dem tatsächlichen Verhalten der beteiligten Lichtwellen nahekommt. In Wirklichkeit werden Lichtwellen, die von einer Lichtquelle durch Participating Media abgestrahlt werden, von unzähligen Partikeln unterschiedlichster Größe entweder abgelenkt oder absorbiert. Dieses als Lichtstreuung (engl. Light Scattering) bezeichnete Phänomen führt beispielsweise dazu, dass, wenn wir eine Taschenlampe auf Rauch oder Wasserdampf richten, das emittierte Licht und

das sogenannte Lichtvolumen für uns visuell wahrnehmbar werden. Die Lichtwellen werden von den Partikeln in diesen Medien ein oder mehrere Male abgelenkt und treffen mit etwas Glück hin und wieder oder sehr häufig die Netzhaut unseres Auges. Falls die Lichtwellen in Richtung des Betrachters abgelenkt werden – und somit zu der totalen Strahldichte beitragen –, wird dies als In-Scattering bezeichnet. Falls die Wellen in eine für den Betrachter unbedeutende Richtung abgelenkt werden, wird dies als Out-Scattering bezeichnet.

Diese Beobachtungen können für die Erklärung unterschiedlichster Naturphänomene herangezogen werden. So lässt sich mit Lichtstreuung beispielsweise sowohl die Farbgebung unseres Himmels, das Erscheinen der sogenannten God Rays oder Strahlenbüschel (engl. Crepuscular Rays), als auch die Interaktion von Bühnenscheinwerfern mit dem von Nebelmaschinen abgesonderten Nebel erklären. Das sind natürlich nur einige wenige exemplarisch gewählt Beispiele.

So viel zunächst zu der grundlegenden Theorie. **Abbildung 1** zeigt einen Teilbereich eines Levels von Lords of the Fallen mit und ohne volumetrische Beleuchtung. Das Sonnenlicht,

Die Serie im Überblick

■ Teil 1

Auf dem Weg in die nächste Generation

■ Teil 2

Into the Guts of the Rendering-Engine

■ Teil 3

Visuelle Effekte für die nächste Generation – Teil 1

■ Teil 4

Visuelle Effekte für die nächste Generation – Teil 2

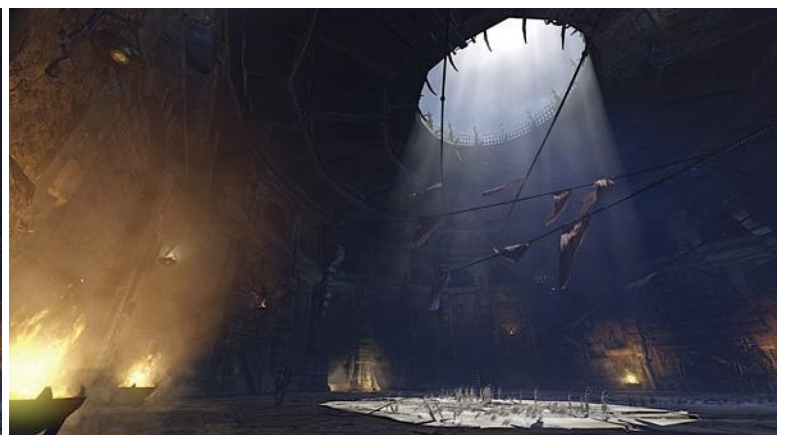


Abbildung 1 Ein Level aus Lords of the Fallen ohne volumetrische Beleuchtung (links) und mit volumetrischer Beleuchtung (rechts). Das Lichtvolumen wird durch ein als Lichtstreuung bezeichnetes Phänomen für den Betrachter sichtbar und gibt diesem das Gefühl, dass die Luft tatsächlich mit unzähligen kleinen Staubpartikeln gefüllt ist.

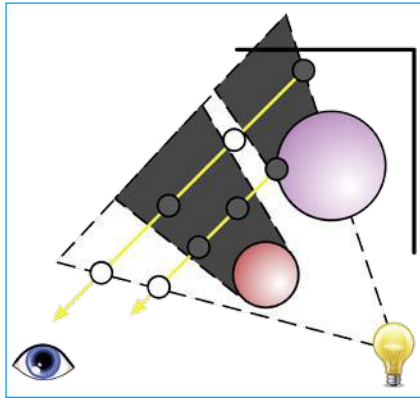


Abbildung 2 Für jedes Licht wird innerhalb der zugehörigen Lichtgeometrie eine bestimmte Menge von Samples genommen; der In-Scattering Term wird damit dann unter Berücksichtigung der Shadow Map evaluiert.

das durch den Ring in der Decke eindringt, wird aufgrund der in der Luft vorherrschenden Staubpartikel gestreut und somit für uns wahrnehmbar. Der Effekt verleiht der Szene den Eindruck, dass es sich nicht um ein vollständig steriles Gebiet handelt, sondern tatsächlich um äußerst staubige und dreckige Katakomben. Wie dieses Beispiel vermuten lässt, wollten wir in *Lords of the Fallen* mithilfe von volumetrischen Beleuchtungseffekten die Interaktion sämtlicher Lichtquellen und Lichtquellentypen mit dunstigen oder nebligen Arealen simulieren. Das System sollte vollkommen dynamisch funktionieren und die Interaktion mit allen Objekten in der Szene unterstützen. Besonders wichtig war uns auch, dass das System bei der Verwendung im Kontext der Playstation 4 und Xbox One kein Performanzproblem darstellt.

Von einer zu vielen Lichtquellen

Die Implementierung der herkömmlichen Beleuchtung unterscheidet sich von Renderersystem zu Renderersystem. Im Zusammenhang mit einem Deferred Renderer, wie er beispielsweise in *Fledge* verwendet wird, wird für jeden rasterisierten Pixel auf dem Bildschirm im Kontext des jeweils betrachteten Lichts die Strahldichte evaluiert und additiv in einem speziellen Puffer akkumuliert. Hierfür wird für jedes Licht eine Geometrie gerendert, die ungefähr dem Lichtvolumen entspricht, das durch die jeweiligen Einstellungen der Lichtquelle entsteht. Für Punktlichter sind dies etwa Sphären, während für Spotlichter eine komplexere Kegelgeometrie zum Einsatz kommt. Innerhalb des gerenderten Lichtvolumens wird die Strahlstärke auf der eingeschlossenen Geometrie der Szene evaluiert. Genau an dieser Stelle haben wir auch unser volumetrisches Beleuchtungssystem integriert.

Die Grundidee basiert auf den Ergebnissen der Forschungsarbeit »Real-time Volumetric Lighting in Participating Media«. Während das ursprüngliche Verfahren nur für eine einzige, direktionale Lichtquelle konzipiert war, funktioniert unser Ansatz mit sämtlichen Lichtquellen und Lichttypen in der gesamten Szene. So evaluieren wir für jede Lichtquelle, die auf

dem Bildschirm dargestellt werden soll, neben der herkömmlichen Strahldichte zudem das »eingestreute« (engl. in-scattered) Licht. Dieses wird für jedes Licht und jede Lichtgeometrie, die wir rendern, zusätzlich mithilfe von Ray Marching und der gleichzeitigen Evaluation der Shadow Map berechnet und in einem zusätzlichen Puffer, dem sogenannten Accumulation Buffer, additiv übereinandergeblendet. Dieses Verfahren ist in einer vereinfachten Form in **Abbildung 2** dargestellt.

Low-Resolution Rendering als erste Optimierung

Die Verwendung von Ray Marching kann auch heute moderne GPUs noch schnell an ihre Grenzen treiben. Das ist wenig verwunderlich, weil bei einer nativen Implementierung für die Berechnung des In-Scattering Terms für ein direktionales Licht beispielsweise häufig mehr als 128 Samples pro Pixel in einem Vollbild-Render-Target abgetastet werden müssen. Je nach Komplexität des gewählten, volumetrischen Effekts ist auch die Anzahl der ALU-Instruktion pro Iteration nicht unbedingt vernachlässigbar. Weil wir volumetrische Beleuchtungseffekte für bis zu 16 Lichtquellen variierender Größe gleichzeitig auf dem Bildschirm ermöglichen wollten, war dies für uns natürlich keine zulässige Lösung. Aus diesem Grund haben wir eine Vielzahl von Techniken eingesetzt, um die Komplexität der Berechnung des In-Scattering Terms auf ein Minimum zu reduzieren.

Der erste und für uns offensichtlichste Schritt war zunächst, die Berechnung des Beitrags pro Licht von einem Vollbild-Render-Target auf ein Render Target mit der Hälfte oder einem Viertel der Größe der gesamten Bildschirmauflösung zu verlagern. Wie bei Partikelsystemen, die dieses sogenannte Low-Resolution Rendering einsetzen, treten auch bei volumetrischen Effekten Artefakte auf, wenn das kleinere Render Target in einem abschließenden Pass über den Frontbuffer geblendet wird. Die Kanten der hochauflösenden Geometrie wirken durch das Blending verwaschen und pixelig. Deshalb verwenden wir an dieser Stelle eine »Nearest



Abbildung 3 Der Gather-Pass ohne (links) und mit bilateralem Filtering (rechts). Hier ist deutlich zu sehen, dass ein Gauss-Filter ohne Einbezug der Tiefenwerte alle feinen Details der Szenengeometrie zerstören kann.

Depth Upsampling« genannte Technik, um in einem finalen Pass das kleinere Render Target des Accumulation Buffers, unter Einbezug der Kanten des hochauflösenden Frontbuffers, auf die finale Größe nach oben zu skalieren.

Einsatz von Interleaved Sampling

Trotz der Verwendung von Low-Resolution Rendering – und der damit global reduzierten Anzahl von zu berücksichtigenden Fragmenten – ist die Gesamtanzahl der notwendigen Samples pro Pixel immer noch immens. Um dieses Problem zu umgehen, machen wir es uns zunutze, dass sich das eingestreuete Licht zwischen benachbarten Pixeln häufig nur geringfügig oder gar nicht unterscheidet. Dies ermöglicht es, das gesamte Ray Marching, das für einen einzelnen Pixel ausgeführt wird, auf 8x8 Pixel große Blöcke zu verteilen. Jeder Pixel eines Blocks evaluiert somit einen unterschiedlichen Teil des gesamten Strahls und wir können beispielsweise mit 16 Samples pro Pixel eine Gesamtanzahl von 1024 totalen Samples erreichen. Das hierdurch entstehende Dithering-ähnliche Muster wird in einem weiteren Pass, von uns als Gather-Pass betitelt, mithilfe eines separablen Gauss-Filters (15 Samples in der horizontalen und vertikalen Richtung haben sich für uns als ideal erwiesen) entfernt. Um zusätzliche Artefakte zu vermeiden, beziehen wir an dieser Stelle mögliche Diskontinuitäten der Tiefenwerte im Tiefenpuffer mit ein, um die Kanten der umliegenden Geometrie zu erhalten. Dieses Verfahren wird häufig als bilaterale Filterung bezeichnet. Dass dieser Schritt tatsächlich notwendig ist, um die visuelle Qualität der gerenderten Szene zu erhalten, zeigt **Abbildung 3**.

Abschließend zeigt **Abbildung 4** das Zusammenspiel sämtlicher Pässe, die für die Berechnung der volumetrischen Beleuchtungsinformationen nötig sind.

Erweiterung des Systems

Während der Entwicklung haben wir uns zahlreiche Möglichkeiten überlegt, um die volumetrischen Effekte zu erweitern und vielfältiger zu gestalten. So können wir zum Beispiel einfache 2D- und 3D-Texturen animieren und dazu mehrfarbige Lichtstrahlen oder großflächige Wolkenschatten bis hin zu Staubeffekten ohne Einbezug der Shadow Map zu simulieren. **Abbildung 5** zeigt die Ergebnisse nach dem Einsatz verschiedener der soeben erwähnten volumetrischen Beleuchtungseffekte.

Um das Erscheinungsbild zu stabilisieren, wenden wir zudem in einem finalen Pass ein Verfahren an, das sich mit der Bezeichnung »Temporale Reprojektion« etabliert hat. Hierbei werden die Ergebnisse des vorherigen Frames mit jenen des aktuellen Frames unter Zuhilfenahme einer speziellen Gewichtungsfunktion geblendet. Als Basis für die Implementierung haben wir den Einsatz des Verfahrens für die Stabilisation des »Screen Space Ambient Occlusion«-Effekts in Battlefield 3 verwendet.

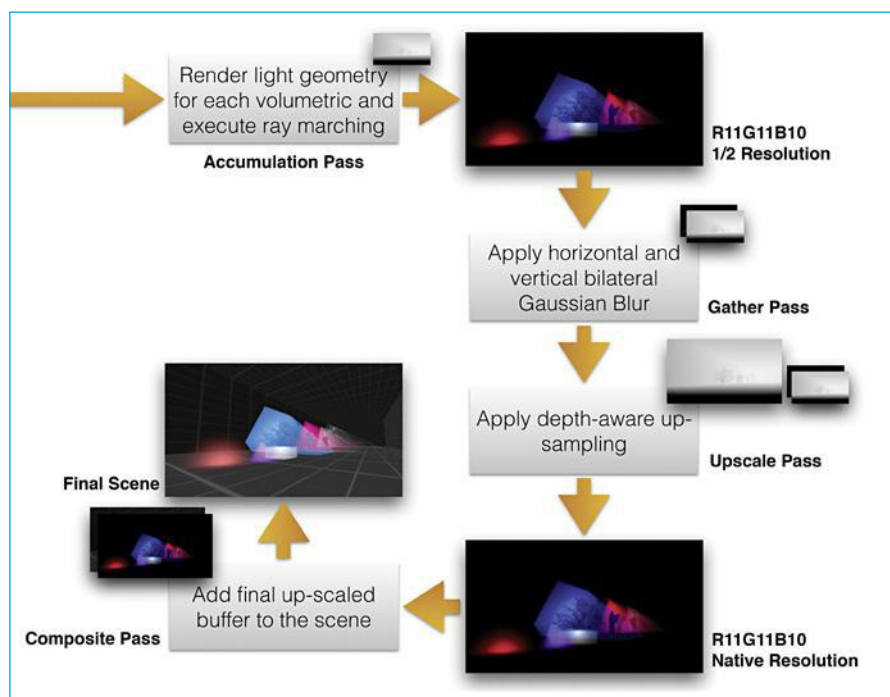


Abbildung 4 Sämtliche Renderpässe, die für die Berechnung der volumetrischen Lichter von Bedeutung sind.

Innerhalb dieses Abschnitts wurden die wichtigsten Subsysteme unseres volumetrischen Beleuchtungssystems vorgestellt sowie einige Lösungsmöglichkeiten für die aufkommenden visuellen und auf die Performanz bezogenen Probleme bei der Implementierung. Wir sind uns bewusst, dass dieser Abschnitt natürlich nur einen sehr kleinen und eingeschränkten Einblick in die Implementierungsdetails des Gesamtsystems bietet – eine detaillierte Beschreibung würde jedoch den Rahmen dieses Artikels deutlich überschreiten. Wir sind uns aber sicher, dass diese Informationen unter Einbezug der erwähnten Referenzen eine sehr gute Grundlage für die Entwicklung eines eigenen volumetrischen Beleuchtungssystems bilden können.

Kameraeffekte auf den nächsten Level heben

Während der Entwicklung von Lords of the Fallen wurden auch die Kameraeffekte der Engine stark überarbeitet und speziell das Vorgehen zur Darstellung von Schärfentiefe von Grund auf überholt. Dieses Verfahren wird allerdings noch nicht im Produktionsbetrieb genutzt und fällt daher unter die Kategorie Research and Development (R&D). Außerdem wurde ein neues, flexibles Lens-Flare-System entwickelt.

Fledge unterstützt ein scatterbasiertes »Depth of Field«-Verfahren, das es ermöglicht, das Bokeh des unfokussierten Bereichs flexibler zu konfigurieren. Die Charakteristik des Bokeh wird häufig mit »busy«, »creamy«, »harsh«, »gentle« oder ähnlichen Begriffen beschrieben, wobei die wahrgenommene Qualität sehr subjektiv ist. In Videospielen zeichnet sich seit einigen Jahren der Trend ab, Formen und Details in unscharfen Bereichen zu erhalten

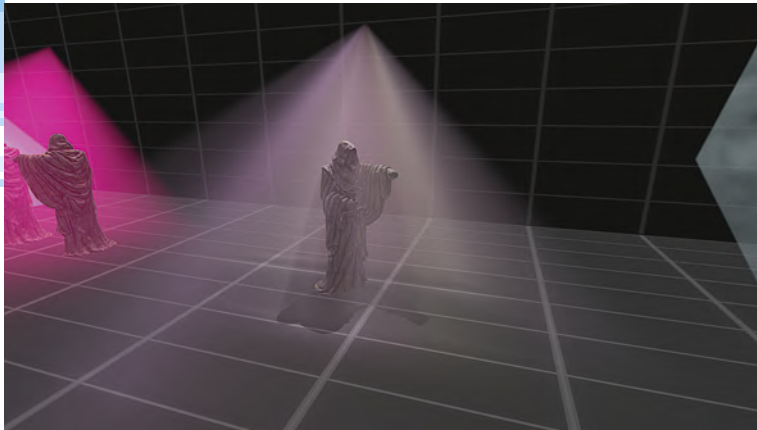
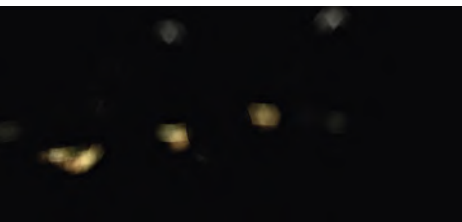
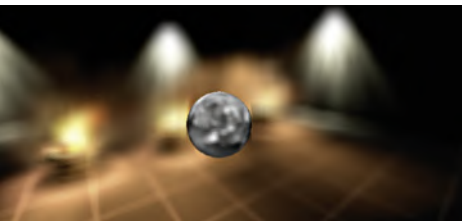


Abbildung 5 Beispiele für verschiedene Effekte, die mit den volumetrischen Lichtern erzeugt werden können. Einsatz einer 2D-Projektions-Textur (links) und einer 3D-Noise-Texture (rechts).



Die einzelnen Teilschritte des Verfahrens: Berechnen des CoC (oben), Gather-DoF (Mitte) und Bokeh Sprite Target (unten).

und ein homogenes Verwischen zu vermeiden. Die Qualität des Bokeh ist schwer messbar und meist auch stark abhängig vom Kontext. Häufig wird allerdings eine Art »creamy«-Bokeh verwendet und versucht, in kontrastreichen Bereichen Details und Highlights zu erhalten, aber auch eine gute Abgrenzung von unfokussierten und fokussierten Bereichen zu gewährleisten. Dies wird heute meist kurz mit »Bokeh Depth of Field« oder Ähnlichem bezeichnet.

Das für Fledge implementierte Verfahren basiert vollständig auf physikalischen Parametern zur Beschreibung der Kamera (F-Stop, Focal Length ...) und wird in mehreren Teilschritten realisiert:

- Erzeugen der CoC-Map
- Point Extraction
- Vertikal/Horizontal Gather-DoF
- Rendern der Bokeh Sprites
- Composition

Erzeugen der Bokeh-Sprites

Das Verfahren erzeugt Sprites (quadratische, zum Bildschirm ausgerichtete Geometrien), um den Bokeh-Effekt zu realisieren. Dazu wird für jeden Pixel der zugehörige »Circle of Confusion« berechnet und die Sprites mit dem errechneten Wert skaliert. Jeder Sprite wird zusätzlich mit der Farbe des zugehörigen Pixels eingefärbt und mit einer Bokeh-Maske gerendert. Die Maske dient zur Simulation beliebiger Formen von Kamerablenden.

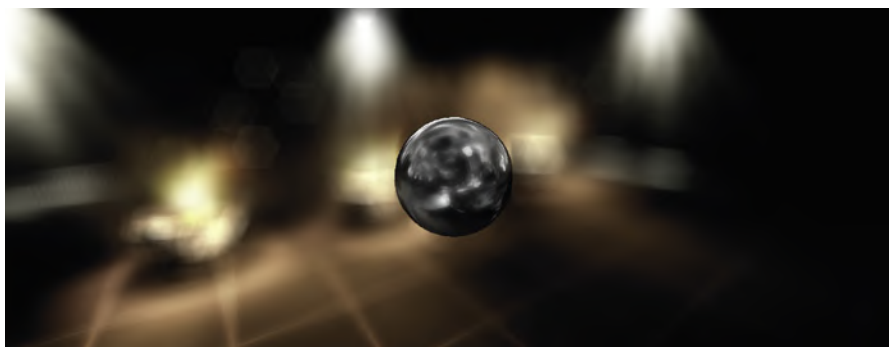


Abbildung 6 Das finale Bild, das sich aus den oben gezeigten Schritten ergibt.

Mit diesem Ansatz können sehr gute Ergebnisse erzielt werden, allerdings kann das Vorgehen aufgrund der hohen Anzahl geschaffener Sprites und dem daraus resultierendem Overdraw schnell zu Performanceeinbrüchen führen.

Mithilfe der Beobachtung, dass Details nur in kontrastreichen Bereichen erhalten werden müssen, kann dieser Schritt weiter optimiert werden. Hierzu werden Sprites nur für »prominente« Bereiche erzeugt und für niedrigfrequente Bereiche ein eigenes Verfahren eingesetzt, das für ein homogenes, weiches Bokeh sorgt und deutlich effizienter berechnet werden kann (wie zum Beispiel ein einfacher Box Blur oder Gaussian Blur, ...). Die Bokeh Sprites werden schließlich in ein eigenes Render Target geschrieben. Der letzte Schritt kombiniert die Szene mit dem Bokeh Sprite Target, was in [Abbildung 6](#) dargestellt ist.

Das von Fledge verwendete Verfahren nutzt DirectX-10/11-Features (Geometry Shader, AppendStructuredBuffer etc.) und konnte damit äußerst intuitiv implementiert werden. Weniger intuitiv kann der Algorithmus allerdings auch für ältere Hardware, allein unter Verwendung von Vertex- und Pixelshadern, realisiert werden.

Lens Flares via Blueprint

Neben der überarbeiteten »Depth of Field«-Implementierung verfügt Fledge über ein komplett neues Lens-Flare-System. In der Planung wurden verschiedene Optionen analysiert. Eine vollständig prozedurale Lösung wurde allerdings aufgrund zu geringer Flexibilität verworfen. Das finale System wird vollständig von unseren Designern angetrieben und hat keinerlei physikalische Basis. Dies erlaubt es allerdings, eine große Anzahl verschiedener Lens-Flare-Effekte, wie etwa anamorphe Lens Flares, zu erzeugen, ohne auf spezielle Lösungen zurückgreifen zu müssen.

Lens Flares werden über sogenannte Blueprints erstellt. Ein Blueprint ist eine Beschreibung des Lens Flare bzw. der einzelnen Linseneffekte. Um Lens Flares in einem Level zu nutzen, bietet Fledge eine eigene Komponente (als `LensFlareComponent` bezeichnet)

an. Somit kann quasi jedes Objekt im Level auch einen Lens Flare erzeugen.

Für teilweise überdeckte Lens-Flare-Quellen ist es oft wichtig, ein weiches Ausblenden der Flares, abhängig vom Grad der Überdeckung, zu gewährleisten. Um dies zu realisieren, konnte auf das Occlusion Culling System der Engine zurückgegriffen werden. Das verwendete System verfügt über einen Software Rasterizer, der Geometrie asynchron und sehr effizient rasterisiert und auf Überdeckung von Objekten testet, um verdeckte Objekte zu cullen. Als Culling werden Verfahren bezeichnet, bei denen Objekte anhand verschiedener Kriterien vom Rendering ausgeschlossen werden - wie beispielsweise der Überdeckungstest im Kontext vom Occlusion Culling oder wenn sich Objekte außerhalb der Sichtperspektive der Kamera befinden.

Für diesen Fall wurde das Occlusion Culling System erweitert, um feststellen zu können, wie viel Prozent eines Objektes auf dem Bildschirm tatsächlich sichtbar sind. Der ermittelte Wert konnte schließlich zum Ausblenden eines Lens Flares verwendet werden. **Abbildung 7** zeigt ein Beispiel für den Einsatz des Lens-Flare-Systems.

Aufbrechen wiederholender Muster im Leveldesign

Bereits relativ früh in der Vorproduktion von Lords of the Fallen war klar, dass die technischen Anforderungen des Spiels einen neuen Workflow zum Erstellen der Levels notwendig machen würden. Wir wollten auf mehreren Ebenen von den Vorteilen eines modularen Ansatzes profitieren: Die Iterationszeiten sollten verringert werden – einerseits durch die Wiederverwendbarkeit einzelner Assets, andererseits durch eine bessere Integration in unsere Tool Pipeline. Aus technischer Sicht war es zwangsläufig notwendig, eine Lösung zu entwickeln, welche die gewünschte visuelle Qualität bei bestmöglicher Performanz gewährleisten konnte. Durch einen modularen Ansatz konnten wir genau das durch einfacheres Batching und verstärkte Nutzung von Instancing erreichen. Die Anforderung für Lords of the Fallen machten allerdings einen »einzigartigen Look« not-



Abbildung 7 Mit der LensFlareComponent von Fledge kann jedes Objekt im Level einen Lens Flare erzeugen.

wendig, der mit sich ständig wiederholenden Bauteilen nicht umzusetzen war. Um diese Problematik zu umgehen, wurden Deferred Decals beziehungsweise Screen Space Decals in die Engine integriert. Neben der Verwendung als Leveldekoration wurden Decals auch an vielen anderen Stellen eingesetzt: Aktuell verwendet Lords of the Fallen solche Decals für Bluteffekte, Fußspuren im Schnee, Brandspuren und vieles mehr.

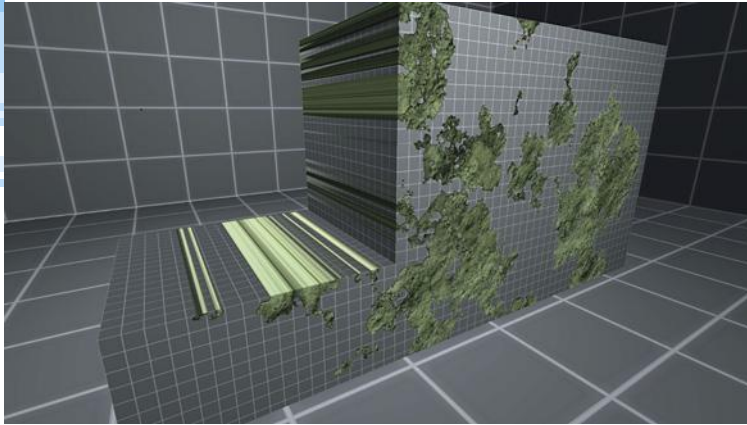
Manipulation des G-Buffers durch Deferred Decals

Deferred Decals integrieren sich perfekt in eine Deferred Rendering Pipeline und können von bereits vorhandenen Teilschritten profitieren. Deferred Decals stellen zum Beispiel eine einfache Lösung dar, um die Oberflächenbeschaffenheit von Objekten zu verändern. So kann ein Nässe-Effekt über die Manipulation der Rauheit einer Oberfläche erzielt werden. Oder die Oberflächenstruktur kann verändert werden, indem die Decals die Normalen überschreiben.

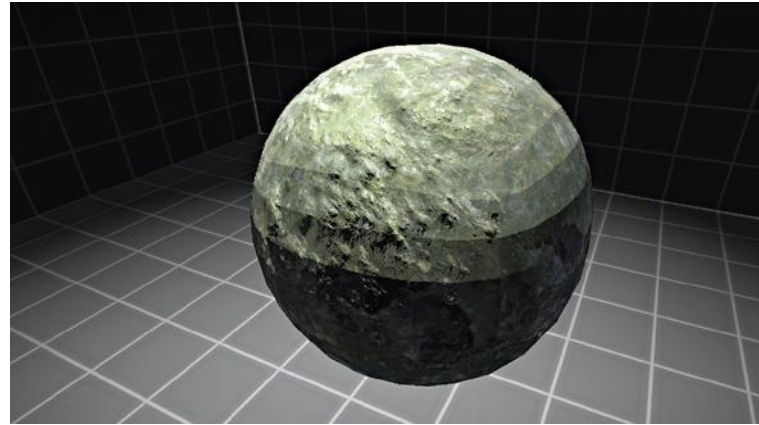
Die Grundidee hinter Deferred Decals ist einfach. Ein umschließendes Volumen wird als Proxy gerendert und für jeden Pixel die lokale



Die Decals werden in Lords of the Fallen beispielsweise auch für Bluteffekte eingesetzt.



Projektionsartefakte eines Decals auf einer parallel zur Projektionsrichtung ausgerichteten Fläche.



Der »Flat Shading Look« aufgrund fehlerhaft rekonstruierter Oberflächen-Normalen.

Position rekonstruiert. Dazu wird der Tiefenpuffer verwendet, dieser muss also bereits vollständig erzeugt worden sein. Dann werden Decals in einem finalen Pass in die Szene geblendet:

- G-Buffer erzeugen und mit Werten befüllen
- Decal Bounding Volume rendern
- für jeden Pixel die Tiefe aus dem G-Buffer lesen
- aus dem Tiefenwert die lokale Position rekonstruieren
- für Pixel innerhalb des Bounding Volumes Werte manipulieren
- Pixel außerhalb des Bounding Volumes verwerfen

Dieser Ansatz bietet gegenüber anderen Methoden zur Darstellung von Decals massive Vorteile, weil er nicht unter Z-Fighting-Problemen leidet und wesentlich besser mit steigender Szenenkomplexität skaliert, als beispielsweise Ansätze die Teile der unterliegenden Geometrie klonen. Daneben existieren auch einige Nachteile, für die allerdings zum Großteil hinreichend gute Lösungen existieren, weshalb in diesem Artikel nicht weiter darauf eingegangen wird.

In vielen Fällen ist es nicht erwünscht, dass Decals auf alle Objekte angewendet werden. Dafür wurde ein einfaches System entwickelt, das Ausmaskieren einzelner Objekte mittels einer Stenciltechnik erlaubt. Ein initialer Ansatz ermöglicht es, Oberflächen mit verschiedenen Flags zu beschreiben (Welt, Boden, statische Objekte etc.). Für jedes Decal musste festgelegt werden, auf welche Oberflächen-Typen es angewendet werden durfte. Eine Gegnermarkierung sollte beispielsweise nur auf Oberflächen, die mit »Boden« markiert waren, erscheinen, wobei Blutspritzer auf die gesamte statische Weltgeometrie angewendet werden sollten.

Obwohl das System sehr flexibel einstellbar war, unterscheidet das aktuelle System zwischen nur zwei exklusiven Objekttypen: statische und dynamische Objekte. Das Reduzieren auf nur zwei Optionen verringert die notwendigen Wechsel der Render States massiv und stellt einen ausreichend guten Kompromiss zur flexibleren Variante dar.

Ein wichtiges Feature der Deferred-Decal-Implementierung in Fledge ist die Möglichkeit der Manipulation des G-Buffers. Im Verlauf der Entwicklung wurden verschiedene Ansätze evaluiert, um eine korrekte Manipulation der betroffenen Oberflächen-Normalen zu erreichen. Die aktuell verfügbare Version unterscheidet zwischen zwei verschiedenen Modi:

- Überschreiben der Normalen
- Blenden der Normalen

Für beide Modi ist es notwendig, den Tangentialraum des betrachteten Punktes der unterliegenden Oberfläche zu ermitteln, um eine korrekte Ausrichtung der Decal-Normale zu erhalten. Der G-Buffer hält unter anderem bereits eine explizite Beschreibung von (Per-Pixel-) Normalen bereit. Sollen die Normalen komplett überschrieben werden, sind die Informationen allerdings unzureichend, weil die unterliegende Oberflächenstruktur immer sichtbar bleibt. Die Rekonstruktion über eine partielle Ableitung der Raumposition des betrachteten Pixels (ddx , ddy) liefert auf nahezu ebenen Flächen ein perfektes Resultat, stark gekrümmte Flächen leiden allerdings unter einem »Flat Shading Look«.

Die vorerst finale Lösung verwendet zur Rekonstruktion der Normalen die Ausrichtung des Decals als Basis. Decal-Normalen werden mit dieser Lösung nicht korrekt um Objekte gelegt; der Ansatz bietet aber die Möglichkeit, problematische Fälle direkt im Editor zu korrigieren.

Abschließende Worte

An dieser Stelle endet der erste Teil unserer Artikel. In der kommenden Ausgabe werden wir einen Einblick in unser CPU- und GPU-basiertes Partikelsystem, das damit verbundene Particle Lighting sowie in unsere auf PhysX und APEX aufbauenden physikalischen Effekte geben. Zudem stellen wir einige prototypische Techniken und Systeme vor, die aktuell noch in Lords of the Fallen angewendet werden. Besonders spannend ist dabei unser auf Compute Shader basierenden Systems zur Simulation und zum Rendering von Haar, Fell und Gras.

David Reinig, Benjamin Glatzel

Bibliografie

Tóth, Balázs & Umenhoffer, Tamás

»Real-time Volumetric Lighting in Participating Media«
<http://sirkan.iit.bme.hu/~szirmay/lightshaft.pdf>

Jansen, Jon & Bavoi, Louis

»Fast rendering of opacity-mapped particles using DirectX 11 tessellation and mixed resolutions«
<https://developer.nvidia.com/sites/default/files/akamai/gamedev/files/sdk/11/OpacityMappingSDKWhitePaper.pdf>

Bavoi, Louis & Andersson, Johan

»Stable SSAO in Battlefield 3 with Selective Temporal Filtering«
http://dice.se/wp-content/uploads/GDC12_Stable_SSAO_In_BF3_With_STF.pdf

Persson, Emil

»Volume Decals«, GPU Pro 2, 2011

Krassnigg, Jan

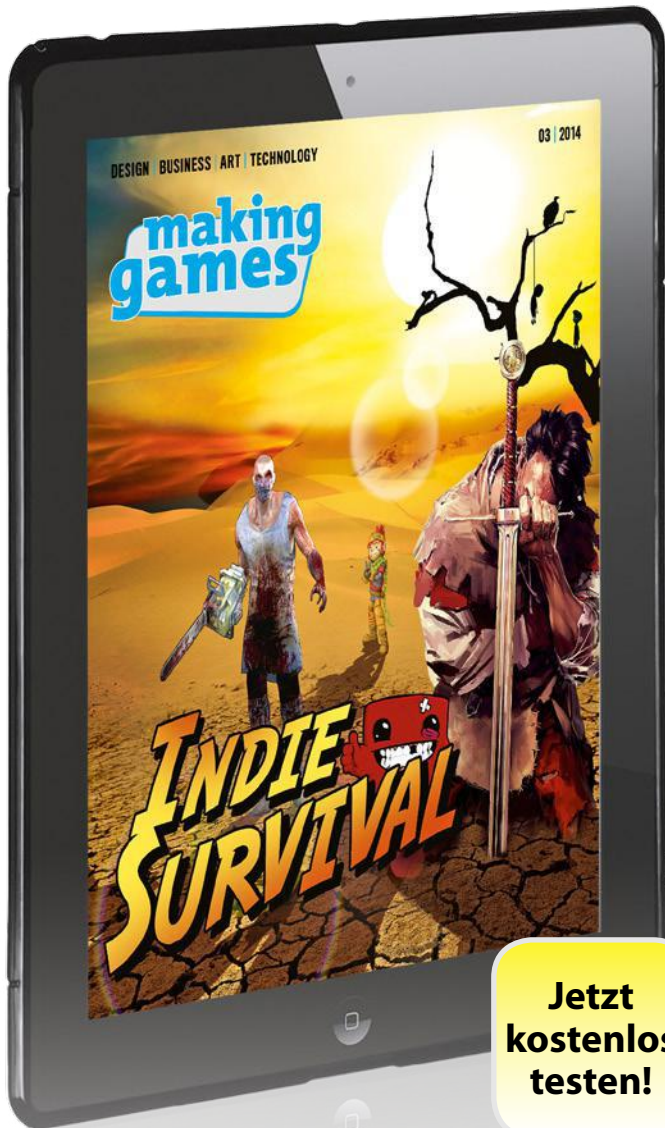
»A Deferred Decal Rendering Technique«
 Game Engine Gems 1, 2010

Kim, Pope

»Screen Space Decals in Warhammer 40,000: Space Marine«
<http://www.popekim.com/2012/10/siggraph-2012-screen-space-decals-in.html>



Jetzt auch für iPad und Android Tablets



**Jetzt
kostenlos
testen!**



Hochwertig

Edles Layout,
optimiert für das mobile Lesen



Interaktiv

Bildergalerien, Videos,
weiterführende Quellen



Preiswert

Für Print-Abonnenten
nur 14,40 Euro im Jahr

**Alle Infos für Neubesteller
und Umsteiger:**

www.makinggames.de/app



www.makinggames.de/appstore



www.makinggames.de/googleplay



DIE ENTWICKLUNG EINER NEUEN MARKE IM MOBILE GAMING

Die Macher des Mobile-Spiels Royal Revolt lernten auf die harte Tour, dass soziale Tools für Mobile-Spiele wichtig sind, um virales Wachstum zu erreichen. Vor allem dann, wenn die Monetarisierung nicht klappt. Diese und andere Erkenntnisse flossen in die Entwicklung des Nachfolgers Royal Revolt 2 ein.



Klaas Kersting
ist Gründer und CEO
von flaregames.

2011 gründete Klaas flaregames, ein Unternehmen für mobile Free2Play-Spiele. In einem früheren Leben hat er den Online-Games-Riesen Gameforge gegründet und als CEO zu einem Unternehmen mit mehr als 100 Millionen Euro Umsatz gemacht. Nebenbei ist er Angel Investor bei einer Reihe von Web- und Games-Unternehmen.

Vorab ein paar historische Hintergründe: Flaregames wurde 2011 als Entwickler gegründet und verbrachte den größten Teil dieses Jahres mit einem überambitionierten Projekt, dem »location-based augmented reality game« namens »Flatmates«. Ende 2011 legten wir das Spiel jedoch auf Eis und konzentrierten uns auf die Entwicklung von F2P-Spielen mit Midcore-Anspruch.

Anfang 2012 hatten wir mehrere Projekte gleichzeitig in der Pipeline, aber alle waren 2D-Spiele. Und allesamt basierten auf dem Corona-Framework, welches zwar eine leichte Parallelentwicklung für Android und iOS erlaubt, aber das Projekt auch einschränkt. Mir ging das alles zu langsam und ich wollte Spiele zukaufen, um schneller mehr lernen zu können. Wir wollten unsere Ideen zum Marketing ausprobieren, unsere Ideen zur Monetarisierung, zum Tracking. Und wir hatten Geld auf dem Konto dank einer erfolgreich abgeschlossenen Finanzierungsrunde.

Also gingen wir auf die Suche nach Entwicklern, um unser Portfolio aufzustocken. Nach einigen Gesprächen war uns klar, dass es nicht leicht sein würde, passende Partner zu finden. Manche brauchten zu viel Leitung und wollten an der Hand gehalten werden, andere zeigten von Beginn an eine gewisse Beratungsresistenz. Und dann trafen wir keen games.

Partnersuche erfolgreich abgeschlossen

Keen games hatte Anfang 2012 den Entschluss gefasst, sich in die Bereiche »Free2Play« und »Mobile« zu wagen. Ideen und Konzepte lagen vor. Zunächst wollten sie selbst publishen, entschlossen sich dann aber doch dazu, sich einen Partner zu suchen. Suchprofil: Expertise in Mobile Games und F2P – und in der Lage, ein Projekt zu finanzieren.

Als wir uns trafen, zeigte sich schnell, dass die Erwartungen zueinander passten: Keen überzeugten uns mit ihrer Multiplattform-Development-Kenntnis, ihrem Qualitätsbewusstsein, ihrer Technologie und ihren

Dev-Pipelines. Wir erlebten sie als zupackende, offene Partner. Und die räumliche Nähe zwischen Frankfurt und unserem Hauptsitz in Karlsruhe ist auch kein Nachteil.

Wir überzeugten Keen im Gegenzug mit unserer Expertise in Sachen Monetarisierung und unserem Willen, uns auf eine langfristige Zusammenarbeit einzulassen, sowie der Tatsache, dass wir ebenfalls ein Entwickler sind – und die Probleme und Möglichkeiten des Developments verstehen. Und, mit den Worten von Keen-Mann Pete Walentin: »Es war klar, dass unser Ansprechpartner Klaas, anders als der BizDevler bei einem Publisher, nicht einfach die Firma verlässt und ein Projekt halbfertig zurücklässt.«

Die Idee und der erste Prototyp

Wir hatten klare Vorstellungen, wie das erste Projekt aussehen sollte: Wir wollten Tower Defense, wir wollten bunte Farben und 3D. Außerdem wollten wir einen vermarktungsfähigen Hauptcharakter, wir wollten Multiplattform, wir wollten eine Monetarisierung ähnlich der von »Army of Darkness« (iOS). Was wir nicht wollten, war Multiplayer, das hätte das Projekt zu groß werden lassen.

Keen kam mit einem sensationellen Prototyp: »Knights« war ein cartooniges, charmanter Tower-Offense-Spiel. Nicht zu Hardcore, aber mit Biss. Lustig, aber doch mit genügend Ecken und Kanten, um erfahrene Spieler nicht abzustößen. Farbenfroh, aber nicht quietschbunt. Klares, schnell erfassbares Spielprinzip – und Spaß beim ersten Anspielen.

Schnelle Produktion

Wir entschlossen uns, schnell in die Produktion zu gehen. Schon im März 2012, nach nur drei Monaten intensiver Prototyp-Phase. Der Softlaunch, quasi die »Beta«, war auf den August angesetzt, der weltweite Release auf iOS sollte Ende September sein. Keen übernahm Code, Audio, Grafik und das Game Design. Wir lieferten Feedback und konzentrierten uns ansonsten auf Marketing, PR und die Beziehungen zu den Plattformhaltern Google und Microsoft. Und selbstverständlich halfen wir bei der Entwicklung der Monetarisierungsmechaniken.

Nach dem Launch waren noch die Fassungen für Android und Windows Phone angesetzt – und natürlich der Support mit neuem Content. Durch ein paar interessante Deals kamen zusätzliche Aufgaben hinzu: Mit der Amazon-Android-Version beispielsweise wurden wir Launch-Partner für den Kindle Fire in Europa. Und zum ersten Geburtstag von Google Play lieferten wir einen »Birthday-Level« und wurden von Google dafür mit einer Promo belohnt.

Die Krux mit der Namensfindung

Während Keen entwickelte, machten wir uns bei flare an die Positionierung. Die erste große Schwierigkeit war der Name. Wir hatten dafür keinen vordefinierten Prozess, die bisherigen Spiele waren allesamt per Bauchgefühl



So sah der erste Startscreen des Prototyps von Royal Revolt aus, der noch den Arbeitstitel »Knights« trug.

benannt worden. Wir entwickelten Matrizen, Namens-Cluster und wälzten in Gesprächen und per Maillisten Hunderte Vorschläge. Kostproben gefällig? Wir hatten charakterzentrische Namen von »Angry Prince« über »Punching Prince« und »Prince Harming« bis zu »Lord Crusher«. Wir hatten allerdings auch auffordernde Namen (»Take the Castles!«), nachahmende Namen (»War of Thrones«), absurde Namen (»Castle Buttkick«) und so weiter. Es war ein schmerzhafter Prozess.

Am Ende kam, nach einer Eingebung eines amerikanischen Game Designers, »Royal Revolt« raus. Leider nicht aussprechbar für Deutsche, was dazu führte, dass viele Fans von der »Royal-Revolt« sprachen, doch zumindest international halbwegs verständlich. Vielleicht nicht der allerbeste Name, aber immerhin brauchbar. Und man wird ohnehin nie erfahren, ob ein anderer Name mehr Downloads produziert hätte.

Das richtige Icon

Die Namensfindung war jedoch nicht das Einzige, was auf Publisher-Seite fehlschlug. Während Keen entwickelte wie eine Maschine, arbeiteten wir am Icon – und hätten es beinahe



Für den ersten Geburtstag von Google Play erstellten wir einen eigenen Level und wurden dafür mit einer Promo belohnt.



Statt des oberen Icons entschieden wir uns nach dem hilfreichen Tipp eines Apple-Mitarbeiters für die untere Variante mit ungewöhnlicher Perspektive.

versaut. Wir hatten eine Reihe von Entwürfen, entschieden uns aber für den falschen. Der Tipp eines Apple-Mitarbeiters brachte uns dann erst zu dem Entwurf, der in der Rückschau ziemlich deutlich der beste ist: starke Farben, sympathisch, ungewöhnliche Perspektive.

Fehlschlag auf der Entwicklungsseite

Auf Entwicklungsseite schlug unterwegs eigentlich nur eine Sache fehl: Beide Seiten vergaßen, dass man in so einem Spiel eigentlich noch eine Motivation braucht, eine Sache, die den Helden antreibt. So entstand in letzter Minute die Story vom Prinzen, der von seinen Verwandten aus dem Königreich vertrieben wird. Und drei Verwandte wurden als Tafeln ins Spiel eingefügt und meldeten sich zwischen den Levels mit Drohungen und Spott. Nun.

Schulterklopfen nach dem Release

Dann kam das Spiel raus, mit guten Ratings der User und großen Features auf beiden Plattformen. Schnell wurde die erste Million Spieler erreicht. Wir hatten ein gutes Spiel rausgebracht! Wir klopfen uns auf die Schultern und dachten wie John »Hannibal« Smith aus der 80er-Jahre Serie »A-Team«: Ich liebe es, wenn ein Plan funktioniert! Die Resonanz der Presse war gut, die Plattformhalter mochten uns. Die Entwicklung mit einem kleinen, agilen Team lief sauber, wir und Keen hatten eine Menge gelernt, alles ist gut. Eigentlich.

Katastrophale Monetarisierung und keine sozialen Tools

Andererseits fiel uns schnell auf, dass wir im Marketing nicht profitabel User kaufen konnten, weil die Monetarisierung eine Katastrophe war. Wir hatten uns zu schnell

für ein Modell und ein Vorbild entschieden und das nicht mehr hinterfragt. Wir hatten unsere Annahmen im Softlaunch nicht mehr vernünftig hinterfragt. Die Tatsache, dass die gesamte Presse unsere faire Monetarisierung lobte, tröstete uns nur wenig.

Und was braucht man, wenn man im Marketing keinen Ansatz findet? Ein Spiel, das sich viral unter den Spielern verbreitet. Leider versäumten wir es eine ganze Zeit lang, Facebook Connect und andere soziale Tools einzubauen – und ließen die User versickern. Millionen User strömten durch das Spiel, ohne Geld auszugeben und andere Spieler einzuladen, einfach weil die Möglichkeiten fehlten.

Die schwarze Null

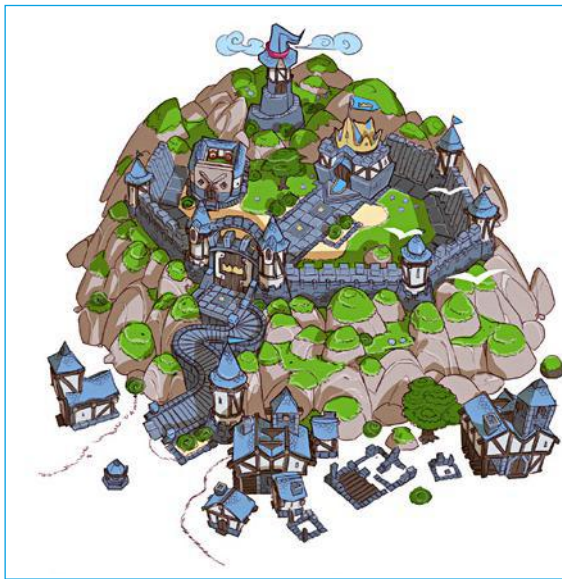
So erreichte das Spiel am Ende mit Mühe und Not eine schwarze Null. Und das auch nur aufgrund des überraschenden Erfolgs, den wir mit Microsofts Hilfe auf Windows Phone und Windows 8 erzielten. Wir gingen die Windows-Plattformen als Letztes an, sogar eine Mac-Version gab es früher. Aber nach heutigem Stand stammt ein Drittel der Downloads von Royal Revolt von Windows-Usern. Ein klassischer Anwendungsfall für die Blue-Ocean-Theorie, nach der man aus umkämpften Märkten ausbrechen soll, in freie Märkte, »blaue Ozeane« eben, nicht die »roten« Gewässer, in denen das Blut der kämpfenden Haie schon das Wasser färbt. Und Windows Phone war für uns eben ein solcher Markt – kleiner als iOS und Android, aber mit weniger Konkurrenz.

Lektionen, die wir gelernt haben:

- Plattformhalter-Beziehungen können über den Erfolg entscheiden.
- Namensgebung ist nichts für zwischendurch.
- Ohne »social features« sollte man sich nicht auf die Straße trauen.



Die Verwandten des Prinzen kamen nachträglich ins Spiel und sollten durch Sticheleien und Drohungen den Ehrgeiz des Spielers wecken.



Der erste Entwurf der Burg für Royal Revolt 2 (links) und der erste Screenshot des Spiels (rechts) in der Gegenüberstellung.

- Singleplayer-Games sind schwer zu monetarisieren.
- Level mit einer Spieldauer von 20 Minuten sind zu lang für Mobile.
- »Content Slicing« ist wichtig: Gib den Spieler nicht zu viel auf einmal.
- Mit nur einer Währung ist Free2Play-Monetarisierung knifflig.

Aber egal, wir würden Royal Revolt keinesfalls missen wollen, denn es hatte den Grundstein für unseren bis dato größten Erfolg gelegt: Royal Revolt 2.

Neuer Anlauf: Royal Revolt 2

Das Briefing für den Nachfolger ergibt sich logisch aus den Lektionen: Multiplayer, Citybuilder-Elemente, Wettbewerb und Ligen, ältere Zielgruppe, komplexere Ökonomie. Alles mit dem Ziel besserer Monetarisierung. Wir begannen die Produktion Anfang 2013 und zielten auf einen Softlaunch im Oktober. Der weltweite Release sollte Ende Februar 2014 sein, mit der Möglichkeit auf schnelle Updates und weitere Versionen.

Unsere Freunde von Keen erweiterten das Spielprinzip von reinem Tower Defense auf einen innovativen Mix aus Tower Defense und Offense, um grundsätzlich eine Balance wie ein »Clash of Clans« zu erreichen, ohne beim Branchenprimus zu sehr abzukupfern. Wir erschufen eine ausbaubare Stadt, einen frei gestaltbaren Defense-Parcours für die Verteidigung und übernahmen für den Angriff auf die Basen anderer Spieler im Wesentlichen das bewährte Spielprinzip des Vorgängers. Beim Prototyping gingen wir neue Wege: Keen baute alles in einen Unity-Prototyp ein, um die grundlegende Funktionalität von Look, UI, Spielprinzip zu beweisen – und übertrug das dann in die eigene Engine, die schon bei Teil eins zum Einsatz gekommen war.

Ein Fokus lag auf der Monetarisierung: Drei Währungen und eine Reihe von Promo-Mechaniken sollten den Spieler dazu bringen, mehr Geld auszugeben. Wir zogen alle Register, um den Kardinalfehler des Vorgängers nicht zu wiederholen.

Parallel entschieden wir, das Spiel grundlegend erwachsener zu positionieren: War der Held im Vorgänger ein Prinz, bekam er jetzt einen martialischeren Look, dunkle Haare und wurde zum König ernannt. Die Signaturfarben wechselten von Blau-Gelb zu Rot-Gelb-Schwarz. Der Hintergrund dieser Maßnahmen: Im ersten Teil hatten wir zu viele junge Spieler – und 16-jährige Jungs sind nicht gerade die bestmonetarisierende Zielgruppe.

Von Beginn an ein Erfolg

Die Entwicklung verlief reibungslos. Wir starteten dann leider doch wieder nicht mit einem vollen Set an Social Features, weil wir sonst den Termin nicht hätten halten können. Und die Tatsache, dass wir kurz vor dem Launch ein paar Dinge gestrichen hatten, verleitete uns, ein leicht überambitioniertes erstes Update zu planen. Aber ansonsten war der Titel vom Start weg ein Erfolg: Die Entwicklungskosten konnten im ersten Monat komplett wieder eingespielt werden. Und wir sind sicher, dass wir Royal Revolt 2 noch lange Zeit profitabel am Leben halten können. Erst kürzlich erschien die Windows-Version, eine ganze Reihe von Updates sind in Planung, die Monetarisierung sieht gut aus – und es gibt auch noch Luft nach oben: In einer bedeutenden Region ist das Spiel noch gar nicht auf dem Markt. **Klaas Kersting**



Royal Revolt 2 wurde erwachsener positioniert: Der Prinz wurde zum König, bekam ein martialischeres Aussehen und die Signaturfarben wurden geändert.

DIE NEUE ENGINE-GENERATION TEIL 1

Eine neue Konsolengeneration und immer leistungsfähigere PCs stellen stetig steigende Anforderungen an Spiele – und damit auch an die Engines. James Golding von Epic Games und David Helgason von Unity Technologies erläutern die Zukunftspläne von Unreal Engine 4 und Unity 5.



James Golding
ist Senior Programmer
bei Epic Games.

James hat seinen Abschluss in Engineering und Computer Science an der renommierten Universität Oxford gemacht. Seit 2002 arbeitet er bei Epic Games, wo er unter anderem an der Entwicklung von »Unreal Tournament 2003« und »Unreal Tournament 2004« sowie den ersten beiden »Gears of War«-Spielen beteiligt war. Anschließend schloss er sich dem Entwicklerteam der Unreal Engine an.

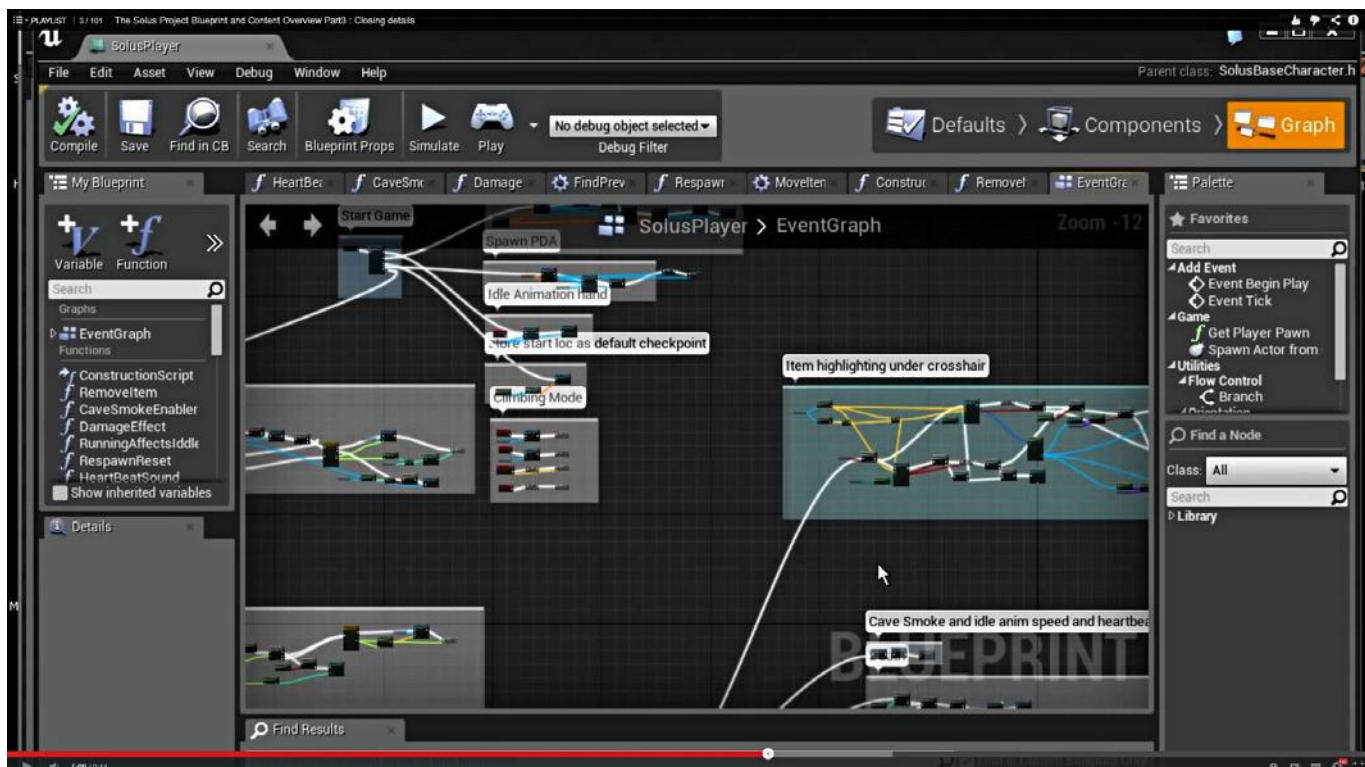
»@EpicJamesG

Making Games Vor dem Interview haben wir unsere Facebook-Community um Fragen für dich gebeten. Sehr oft sind dabei die Themen Support und Anleitungen aufgetaucht. Hast du irgendwelche Tipps für Indie-Entwickler, die zum ersten Mal mit der Unreal Engine 4 arbeiten?

James Golding Anleitungen sind uns auf jeden Fall wichtiger als früher. Wir haben viele Leute, die bisher immer nur frei für uns gearbeitet haben, mittlerweile fest angestellt. Allein für die Dokumentation haben wir jetzt vier bis fünf Vollzeitbeschäftigte.

Making Games Und wie steht es um den Support für eure User?

James Golding Der ist bei uns breiter verteilt. Die meisten arbeiten aber in unserem Hauptsitz in North Carolina. Neben schriftlichen Anleitungen haben wir auch Tutorial-Videos gedreht. Stan Parish, der auch für unsere Trailer zuständig ist, hat bis jetzt schon über 50 Videos zur Unreal Engine 4 gemacht und wird auch nach dem Launch weiterproduzieren. Da kommt es dann darauf an, was die Leute besonders interessiert. Wir haben zum Beispiel eine Reihe über das Bauen von Levels gemacht und eine über Blueprints. Ich habe in letzter Zeit vor allem Templates produziert, quasi als Beispiel für die verschiedenen Genres. Wir haben bereits ein 1st- und 3rd-Person-Template, eins in der Art von



Epic Games gibt Entwicklern unterschiedliche Hilfestellungen zur neuen Engine an die Hand: »Neben schriftlichen Anleitungen haben wir auch jede Menge Tutorial-Videos gedreht.«

»Diablo«, ein Sidescroller- und ein Plattformer-Template veröffentlicht. Auch davon wird es nach dem Launch noch mehr geben.

Making Games Sind das quasi Gratisdemos, mit denen man herumspielen kann?

James Golding Streng genommen sind es nicht mal Demos. Man kann sie benutzen, wenn man mit einem Spiel anfängt. Aber eigentlich zeigen sie nur die Kernelemente für jedes Genre.

Making Games Also wie eine Word-Vorlage?

James Golding Genau. Ein bisschen so, wie wenn man die Vorlage für ein Bewerbungsschreiben aufruft. Außerdem haben wir neben C++ auch Blueprints. Wenn ich also ein Programmierer bin, arbeite ich mit C++. Wenn ich Designer bin und vom Programmieren keine Ahnung habe, kann ich die Blueprints nutzen. Viele unserer Programmierer schreiben auch für unseren Blog. Dadurch haben die Benutzer mehr oder minder direkten Kontakt zu ihnen und können Erfahrungen austauschen. Tim Sweeney, unser Engine-Guru, ist ebenfalls dabei und erklärt verschiedene Nutzungsmöglichkeiten. Unreal-engine.com wird so ständig aktualisiert. Außerdem veranstalten wir Livestreams auf Twitch, in denen wir zum Beispiel das Animationssystem und dessen technische Besonderheiten erklären.

Making Games Kann jeder darauf zugreifen?

James Golding Es ist für jeden zugänglich. Unser Forum ist bereits gut gefüllt, und viele unserer Artists und Entwickler sind aktiv dabei. Bei uns ist jeder mitverantwortlich. Wir haben zwar ein paar Leute im Team, die in Vollzeit für das Forum zuständig sind, aber viele von uns machen es in ihrer Freizeit. Einfach, weil dieses Projekt unser Baby ist. Da steckt so viel Leidenschaft drin und wir wollen, dass es Erfolg hat. Deswegen wollen auch alle wissen, was im Forum vor sich geht. Es gibt sogar ein Wiki und diverse Entwickler-Sessions auf der ganzen Welt. Wir möchten so direkt wie möglich mit der Community zusammenarbeiten.

Making Games Wenn ich vorher mit anderen Engines gearbeitet habe: Was wäre ein guter Einstieg in die Arbeit mit Unreal?

James Golding Das kommt ganz darauf an, was für ein Typ du bist. Für Leute, die sich lieber erst mal genau informieren, haben wir die Videos und die schriftlichen Anleitungen. Wenn du direkt loslegen willst, kannst du dir einfach eines der Templates schnappen und anfangen. Entweder du nimmst sie auseinander, oder du schaust, wo die Unterschiede zu anderen Engines liegen. Auch dabei erfährt man eine ganze Menge. Eigentlich geben wir dir einen Haufen Assets an die Hand und lassen dich damit machen, was du willst. Wir haben versucht, jede Herangehensweise so einfach wie möglich zu gestalten. Wenn du also bei irgendwas feststeckst, werden wir versuchen, das für dich hinzubiegen.

Making Games Mit der Unreal Engine 4 erlaubt ihr jetzt jedem Entwickler Zugriff auf euren Quellcode. Wie funktioniert das genau?

Die Serie im Überblick


■ Teil 1

Unreal Engine 4 und Unity 5

■ Teil 2

CryEngine und Havok

»Bei uns ist jeder mitverantwortlich.«



RANGE-BASED FOR LOOPS

James Golding on May 12, 2014 | Features, Learning, Programming

In the Unreal Engine 4 code base, we have started taking advantage of some of the cool features added in C++11. We are cautious when adopting new language features because we target a wide range of platforms with varying compiler support. One of my favorite new features though is 'range-based for loops', which now works on Unreal's TArray, TMap and TSet containers. Here is an example of how this can really clean up common game code. Previously if I wanted to iterate over an array of AActor* pointers, it used to look like:

```
TArray<AActor*> Actors;
for (int32 ActorIndex=0; ActorIndex<Actors.Num(); ActorIndex++)
{
    AActor* Actor = Actors[ActorIndex];
    Actor->SetActorLocation(NewLocation);
}
```

But using the 'range-based for loop' syntax it now looks like:

```
TArray<AActor*> Actors;
for (AActor* Actor : Actors)
```

GET UNREAL

FULL ENGINE AND SOURCE
€19/MO + 5%

- NEW USER FAQ >
- SUPPORT >
- TUTORIALS >
- DOCUMENTATION >
- GITHUB >
- FORUMS >
- ANSWERHUB >
- WIKI >

GET STARTED

- Blueprint Editor Tips and Tricks
Mar 26, 2014
- Creating A Localization Ready Game in UE4: Part 1-Text
Apr 29, 2014
- Blueprint basics
Mar 21, 2014

05-13-2014, 01:25 PM

Michael Noland Unreal Engine Developer
Join Date: Mar 2014
Location: Epic Cary
Posts: 150

It's a hierarchy. The World you see in your list above contains Level objects, which in turn contain Actor objects, some of which may be StaticMeshActor or Blueprint instances that in turn contain a StaticMeshComponent. However, none of them will contain UStaticMesh objects since those are assets stored in a .uasset, rather than a .umap (the components may reference an asset, but won't contain it; the one exception being lightmap textures which are kind of like assets but return false for IsAsset because they are embedded)

What's your goal? There may be better ways to find or work with all actors in a level, either interactively or at runtime, and there are certainly better ways to just find all content referenced by a set of objects (one of the reference collector archivers for example).

Cheers,
Michael Noland

Last edited by Michael Noland; 05-13-2014 at 01:38 PM

05-14-2014, 05:37 AM

Ken Dahm Junior Member
Join Date: May 2014
Posts: 2

I am writing a custom renderer and therefore I would like to access the geometry/transformations/lights etc.

I thought the best way would be to extract all the data from an *.umap file and feed it to my renderer. Do you know a better way to achieve this goal or is it possible directly render into the editor's viewport?

Thanks for your answer/help.

05-14-2014, 06:01 AM

birdfree@yahoo Junior Member
Join Date: May 2014
Posts: 19

Well since you have obtained UE4 you have full source code access and hence you can modify the renderer or create a new one without hacking.

Neben den Blogbeiträgen der Entwickler ist das Team auch im Forum aktiv: »Wir haben zwar ein paar Leute im Team, die in Vollzeit für das Forum zuständig sind, aber viele von uns machen es in ihrer Freizeit.«

»Ein einfaches Spiel zu machen, muss nicht automatisch 8-Bit-Pixel-Grafik bedeuten.«

James Golding Wir benutzen GitHub, einen Hosting-Dienst für Softwareentwicklung. Von unserer Engine gibt es zwei Versionen: Einerseits die standardisierte, veröffentlichte Version, die man bekommt, wenn man unseren Installer benutzt. Andererseits gibt es noch die Version, die mitten in der Entwicklung steckt. Die kannst du natürlich nicht für dein Spiel verwenden, aber du kannst sehen, was in der Entwicklung passiert. Quasi eine Operation am offenen Herzen. Du kannst sehen, wie wir bei Epic arbeiten, woran wir arbeiten. Falls wir also einen Bug repariert haben, den du auch in deinem Spiel hast, kannst du unseren Bugfix direkt in dein Spiel integrieren. Dafür erstellst du auf GitHub eine sogenannte Gabel mit einer Kopie des Codes, an der du herumbasteln kannst. Oder du erstellst deine eigene Version der Engine, passt sie deinen Anforderungen an und baust sie ein. Wenn du der Meinung bist, was richtig Cooles gemacht zu haben, und das mit anderen teilen möchtest, erstellst du ein Pull Request. Dann können andere Entwickler und auch wir bei Epic auf deine Veränderung zugreifen und sie gegebenenfalls in die Engine einfügen.

Making Games Wie wollt ihr forcieren, dass die Entwickler ihre Modifikationen auch tatsächlich austauschen?

James Golding Ehrlich gesagt weiß ich das nicht. Wir stehen mit unserem Konzept ja noch ganz

am Anfang. Wir werden versuchen, auf den unterschiedlichen Kommunikationskanälen so präsent wie möglich zu sein. So können wir auch überblicken, was die Leute mit dem Code anstellen. Gabeln sie ihn vielleicht nur oder haben sie Probleme beim Schreiben? Am Ende werden wir sehen, wie die Community reagiert. Wir haben uns bemüht, einen sauberen, gut verständlichen Code zu schreiben. Sollten Probleme auftauchen, werden wir uns darum kümmern.

Making Games Vor der Pressekonferenz auf der GDC war Epic immer sehr verschwiegen. Man durfte nicht filmen, es gab sehr strenge NDAs. Jetzt hieß es: »Filmt unbedingt, twittert, facebookt, hier ist der Code für unser Wi-Fi.« Hat die neue Herangehensweise an euer Produkt auch Auswirkungen auf eure Firmenkultur?

James Golding Auf jeden Fall! Eigentlich könnte man das Ganze Epic 4.0 nennen. Für viele von uns, die schon länger dabei sind – ich arbeite hier seit zwölf Jahren – hat Epic 4.0 viel von Epic 1.0. Früher haben wir uns intensiv mit unseren Usern beschäftigt, Werkzeuge und Mods veröffentlicht und uns über Änderungsvorschläge gefreut. Jetzt, wo wir nicht mehr durch einen Publisher agieren, wollen wir da wieder hin. Wir wollen weg vom Elfenbeinturm und wieder hin zum direkten Kontakt mit den Leuten.

Making Games War die Veränderung einfach? Oder bis ins kleinste Detail geplant?

James Golding Einfach war sie sicher nicht. Intern haben wir bereits eine ganze Weile darüber nachgedacht, aber es gab auch viel zu berücksichtigen. Wie sprechen wir die Leute an? Wie werden wir unserer Community gerecht? Auf lange Sicht ist nämlich genau das wichtig. Wir haben uns entschieden, eine offene Firmenpolitik zu verfolgen, weil wir glauben, dass das richtig und wichtig ist. Natürlich ist das Ganze etwas einschüchternd. Aber Epic, vor allem das obere Management, hat uns ermutigt, in den offenen Dialog zu gehen. Mit der Presse zu reden und einzuräumen, dass es Probleme geben wird. Aber trotzdem werden wir zu unserer Idee stehen. Ehrliche Kommunikation ist sehr wichtig.

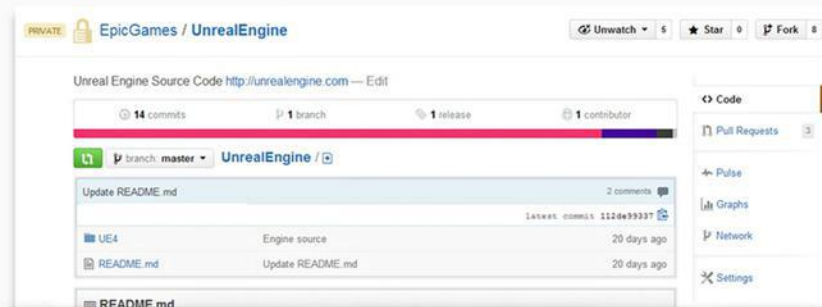
Making Games Wie lange habt ihr denn schon darüber nachgedacht?

James Golding Wahrscheinlich so etwa zwei Jahre. Die neue Art des Spieleentwickelns, der Wandel hin zum direkten Umgang mit den Endnutzern und das Self Publishing – all das hat dazu beigetragen, dass wir unsere Grundsätze in Frage gestellt haben. Und dann überlegt haben, was für Auswirkungen das auf unser Tagesgeschäft haben kann und muss.

Making Games Mobile Gaming wird immer wichtiger. Die Unreal Engine ist dort aber bisher noch nicht sehr präsent. Was habt ihr in diesem Feld vor, wo steht Mobile Gaming im Moment eurer Meinung nach und wie verhält es sich mit dem Unterschied zur Konsole?

WHAT IS GITHUB?

GitHub is a widely-trusted web-based hosting service for software development projects. The entire Unreal Engine source code is available on our GitHub page, ready for you to access!



HOW DO I ACCESS UNREAL ENGINE 4 C++ SOURCE CODE VIA GITHUB?

Just follow these simple steps:

- 1 GitHub**
Sign Up For GitHub
Create an account at [GitHub.com](https://github.com)
- 2 GITHUB ACCOUNT:**
GitHub Account Name
Go to **your account** and associate your GitHub username by entering it into the GitHub account field.
- 3 GitHub.com/EpicGames**
Get started by going to our **GitHub page** and downloading the full source.

DO I NEED SOURCE VIA GITHUB?

Most developers will have everything they need using the installed version of the engine, but if you want to modify the engine directly, or contribute your changes to the community, this is the right place for you.

Via GitHub können Entwickler ihre eigene Version der Unreal Engine anderen Entwicklern zur Verfügung stellen.

James Golding Wir wollen auch im mobilen Sektor präsenter werden. Allerdings eher auf aktuelleren Geräten, wie dem Nexus 5 oder dem iPhone 5S. Unsere Engine eignet sich wunderbar für Mobile Gaming. Die HDR-Nachbearbeitung für Spiegelungen, das Base Shading – all das kann auf den heutigen Geräten voll ausgeschöpft werden. Und es muss ja auch kein 10-Millionen-Dollar-Spiel sein. Mit der Unreal Engine hast du jetzt die Möglichkeit, die Kapazitäten der aktuellen Geräte auszuloten. Die neue, bessere Hardware bringt da nur Vorteile.

Making Games Werden die Spiele dann auch komplexer? Immerhin werden die Entwicklungszyklen immer länger.

James Golding Nicht unbedingt. Ein einfaches Spiel zu machen, muss nicht automatisch 8-Bit-Pixel-Grafik bedeuten. Spiegelungen, Partikel oder Lense Flares sind mittlerweile genauso unkompliziert einzusetzen. Hier kommt die Qualität von der Technik und nicht von einem riesigen Art-Team. Die Kompatibilität mit iOS und Android sollte kein Problem sein, und wir bemühen uns, den Übergang so flüssig wie möglich zu gestalten. Wir sind schon sehr gespannt, was dabei herauskommt, wenn Coding-Kenntnisse keine Voraussetzung mehr sind, um ein Spiel zu erstellen. Ich freue mich auf Ideen, an die bis jetzt noch keiner gedacht hat.

Making Games Und Browsergames? Ihr arbeitet mit Mozilla zusammen. Allerdings kämpft ihr noch mit extrem lange Ladezeiten. Wie lange wird es dauern, bis Unreal eine echte Alternative für Browsergames wird?

James Golding Was das angeht, waren wir von Anfang an sehr ehrlich. Das HTML existiert, es funktioniert irgendwie, doch es ist noch in Arbeit. Wir möchten aber, dass es eine tatsächliche Verwendungsmöglichkeit wird, auch wenn wir im Moment noch einige Probleme damit haben. Glücklicherweise treten ähnliche Probleme ebenfalls bei den Umsetzungen auf der Xbox One und der PS4 auf, sodass ein größeres Team am gleichen Problem arbeitet. Wir bekommen dabei auch Unterstützung von Mozilla.

Making Games Zum Thema Next-Gen-Konsolen: Mit der dritten und vierten Veröffentlichungswelle werden auch die neuen Konsolen für Indie-Entwickler interessanter. Wie sieht es denn hier mit dem Ausschöpfen der Kapazitäten aus? Gibt es große Performanceunterschiede zwischen den Geräten? Und sollten Entwickler auf etwas im Besonderen achten, wenn sie ihr Angebot auf die Konsolen ausweiten wollen?

James Golding Also eigentlich haben sowohl PS4 als auch Xbox One ordentlich was drauf. Zwar hat die PS4 in manchen Punkten die Nase vorn, aber größtenteils sind sie sich ziemlich ähnlich. Obwohl sich die Prozessoren in ihrer Bauweise ähneln, sind die Spiele selbstverständlich nicht ohne Portierungsarbeit übertragbar. Damit wird man sich auch im weiteren Entwick-

lungsverlauf der beiden Konsolen beschäftigen müssen. Man sollte aber im Auge behalten, dass Veränderungen hier kontinuierlich und ziemlich schnell stattfinden. Wie bei der letzten Generation werden die Spiele mit der Zeit immer besser aussehen; die Entwickler müssen ihre Engines ja ebenfalls modifizieren, um alles aus der Hardware herauszuholen zu können.

Making Games Wenn sich die Engines mit den Konsolen weiterentwickeln, werden die Unterschiede zwischen den Konsolen dann irgendwann ganz verschwinden?

James Golding Die Unterschiede bei der Hardware werden nie ganz verschwinden. Es ist allerdings schwer zu sagen. Betrachtet man die Hardware zusammen mit der Software, könnte das natürlich sein.

Making Games Was geht mit der Unreal Engine 4 und den Next-Gen-Konsolen, was vorher noch nicht möglich war?

James Golding Wenn man eine Szene entwirft, sind Lichteinfall, Spiegelungen, Schatten und alles andere einfach sehr kohärent. Neben diesen offensichtlichen Punkten sind es aber auch die kleinen Details. Besonders stolz sind wir beispielsweise auf unsere Treppeneffektglättung. Man investiert viel Zeit in Kleinigkeiten und freut sich dann umso mehr, wenn alles gut zusammenpasst. Wenn dann am Ende ein Setting herauskommt, in das man Gegenstände nur einsetzen muss und alles stimmt, dann wirkt das Ganze direkt viel greifbarer.

Making Games Und das Hardcoding fällt weg?

James Golding Genau. Wir haben vieles optimiert. Was meiner Meinung nach übrigens zu den beeindruckendsten Veränderungen gehört: Es ist nicht mehr so mühsam, Sachen visuell ansprechend oder komplex aussehen zu lassen, weil die Hardware viel besser ist als früher.

Making Games Welche Art von Spiel würdest du in Zukunft gern sehen? Was ist deiner Meinung nach bisher vernachlässigt worden?

James Golding Ich würde gern ein Spiel ohne all die üblichen Verdächtigen der Unreal Engine sehen. Bis jetzt gibt es hauptsächlich Spiele mit einer statischen Umgebung, wenigen Charakteren und düsteren Farben. Ziemlich viele Korridor-Shooter also. Weil die Leute denken, dass die Unreal Engine nichts anderes kann. Deswegen wäre es super, Spiele mit dynamischer Umgebung zu sehen, mit strahlenden, satten Farben, einer Menge Charaktere und Leben. Ich glaube, dass die Engine das ohne Probleme leisten kann. Die Unreal Engine 4 ist auch viel schneller als ihre Vorgängerin. Normalerweise sind Nachfolger komplexer und damit komplizierter, aber dieses Mal ist eine schnellere, gründlichere Engine dabei herausgekommen. Ich würde mich freuen, wenn die Leute ihrer Fantasie freien Lauf lassen. Ich hätte zum Beispiel nichts gegen ein farbenfrohes, rundenbasiertes Strategiespiel.



Bis auf wenige Punkte sind sich die Playstation 4 und die Xbox One ähnlich: »Die Spiele sind aber selbstverständlich nicht ohne Portierungsarbeit übertragbar.«

»Man investiert viel Zeit in Kleinigkeiten und freut sich dann umso mehr, wenn alles gut zusammenpasst.«



Shooter wie »Deadfall Adventures« zählen zu den üblichen Verdächtigen der Unreal Engine: »Die Leute denken fälschlicherweise, dass die Engine nichts anderes kann.«



David Helgason
ist Mitgründer und CEO
von Unity Technologies.

David war als Programmierer bei Egebert Interactive Communication, als Entwickler bei Netropolis und als Business Consultant bei seiner eigenen Firma Panmedia ApS tätig. 2003 gründete er gemeinsam mit Nicholas Francis und Joachim Ante Unity Technologies. Außerdem war er von 2008 bis 2009 Chairman bei Unity Studios und gehört seit 2012 dem Board of Directors von Plain Vanilla Productions an.

@davidhelgason

»Wir haben uns vor langer Zeit für ein Geschäftsmodell entschieden, das Maßstäbe in der Branche gesetzt hat.«

Making Games Vor dem Interview haben wir unsere Facebook-Community gefragt, was sie unbedingt über Unity 5 wissen möchte. Unseren Fan Freddy Blub interessiert besonders, ob es in Zukunft einen stärkeren Fokus auf 2D geben wird – oder dreht sich alles nur noch um 3D?

David Helgason Bei der Unity-5-Ankündigung auf der GDC haben wir zahlreiche neue Features und neue Technologien vorgestellt. Unter anderem gehören dazu auch tolle 2D-Physics-Updates, um den Entwicklern die Arbeit zu erleichtern. Das ist aber nur der erste Schritt. Wir gehen aktuell davon aus, dass jedes Update – oder zumindest die meisten Updates – auch Neuerungen für die 2D-Engine beinhalten werden. Wir haben sie erst im November 2013 auf den Markt gebracht, aber sie ist jetzt schon unheimlich gut. Darauf wollen wir aufbauen.

Making Games 2D ist für euch also immer noch wichtig?

David Helgason Absolut. Grob geschätzt sind rund die Hälfte der Titel, die mit Unity gemacht werden, 2D-Spiele. Vielleicht sind es auch 40 oder 60 Prozent, irgendwo in dieser Größenordnung bewegt sich die Zahl auf jeden Fall. Es sind also eine Menge Spiele, und das ist uns selbstverständlich sehr wichtig.

Making Games Sebastian Kreutz von Frame6 möchte wissen, wie es mit dem Licensing Deal für die Xbox One und die Playstation 4 vorangeht.

David Helgason Unity ist für die Xbox One und die Wii U komplett kostenlos. Bei der Playstation müssen sich die Entwickler zuerst mit Sony un-

terhalten. Wir haben Partnerschaften mit allen drei Unternehmen, aber im Moment sind sehr viele Spiele in der Entwicklung. Wir haben vor kurzem gehört, dass die Hälfte der Playstation-4-Spiele, die gerade in Japan entwickelt werden, auf Unity basieren. Mit »Max: The Curse of Brotherhood« ist auf der Xbox One auch schon ein Spiel erschienen, das mit Unity gebaut wurde. Und viele weitere Titel sind noch in der Entwicklung und in unterschiedlichen Beta-Stadien.

Making Games Shinsen Myamoto fragt: Wird der Support für Oculus Rift in naher Zukunft in der kostenlosen Version zur Verfügung stehen?

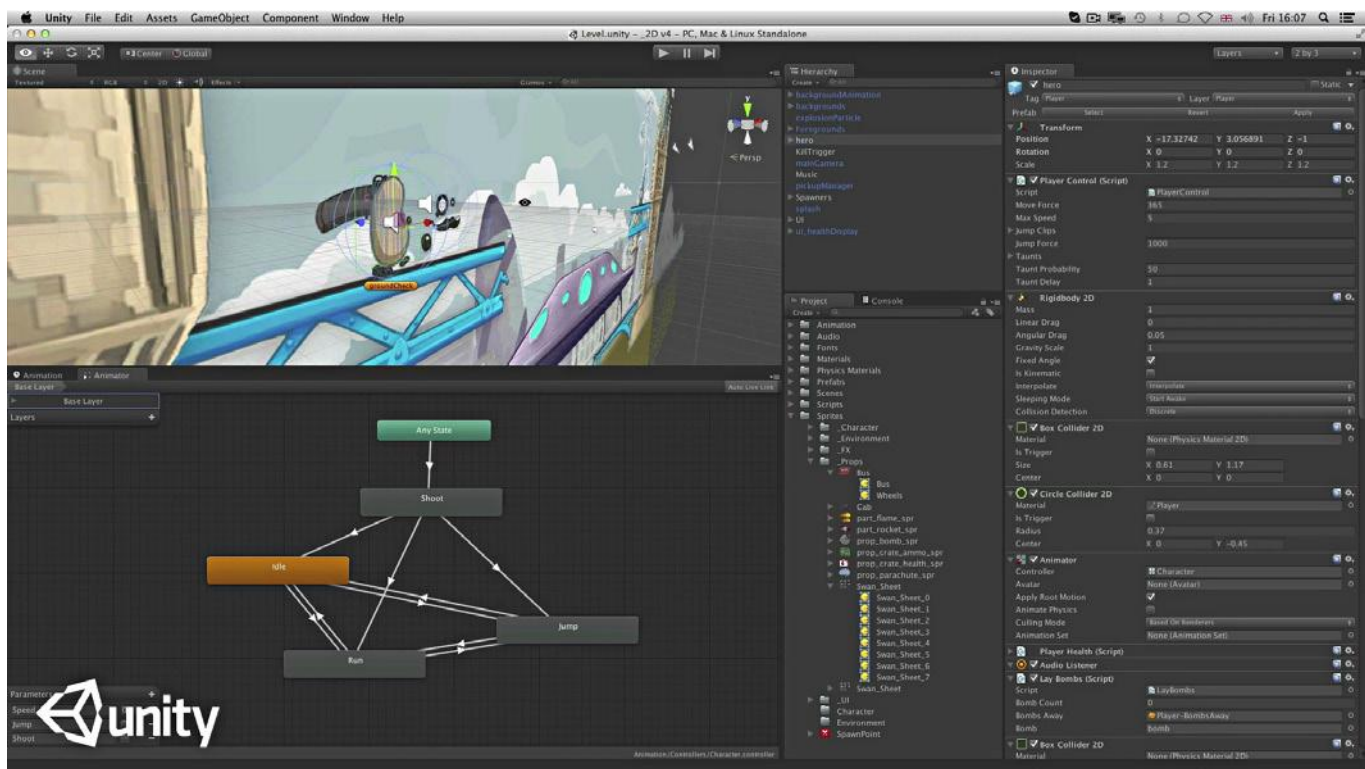
David Helgason Gute Frage. Wir haben tatsächlich eine Partnerschaft mit Oculus – wer ein Dev-Kit für die Rift kauft, bekommt dazu vier Monate Zugang zu Unity Pro. Dieses Angebot wird auch von vielen Entwicklern genutzt.

Making Games Kannst du schon etwas zu »Project Morpheus« sagen? Wird das für euch in Zukunft auch ein Thema sein?

David Helgason Auf jeden Fall. Unsere Partnerschaft mit Sony umfasst alle Geräte, also auch »Morpheus«. Der volle Support für Unity wurde auch in der offiziellen Pressemitteilung erwähnt. Wir wussten also schon eine Weile, dass sie daran arbeiten.

Making Games Pascal Jacob möchte wissen, wann es endlich GPU Instancing und Nested Prefabs geben wird.

David Helgason Ausgezeichnete Frage. Ich habe ehrlich gesagt keine Ahnung. Wir arbeiten daran, aber wann wir fertig sind, kann ich nicht sa-



Rund die Hälfte der mit Unity erstellten Spiele sind 2D-Titel: »Die meisten Updates für Unity 5 werden auch Neuerungen für die 2D-Engine beinhalten.«

gen. Wir legen uns erst auf Releases fest, wenn wir uns sicher sind, dass alles funktioniert. Dieses Maß an Sicherheit haben wir jetzt bei Unity 5.0. Natürlich wird es auch wieder 5.1, 5.2 und so weiter geben, aber wir wissen noch nicht, was diese Versionen alles beinhalten werden.

Making Games Mikhael Repovský fragt, warum man Schatten und Fading zwischen Szenen nicht in der kostenlosen Version nutzen kann.

David Helgason Wir haben uns vor langer Zeit für ein Geschäftsmodell entschieden, das in zwischen Maßstäbe in der Branche gesetzt hat. Wir waren nie vollkommen glücklich damit, aber es hat funktioniert. Allerdings bedeutet das eben, dass einige Nutzer nicht das bekommen, was sie wollen. Das ist frustrierend, denn als Entwickler will ich den Leuten eigentlich ja genau das geben, was sie haben möchten. Auf diese Frage gibt es eigentlich keine gute Antwort. Ich kann nur sagen, dass wir immer wieder darüber nachdenken, wie wir unser Geschäftsmodell in Zukunft verändern könnten.

Making Games Gianluca Pandolfo nutzt Unity auch für animierte Kurzfilme. Ihn interessiert, ob es für dieses Gebiet zukünftig Verbesserungen geben wird, beispielsweise Plugins wie Mixamos Face Plus, neue Post-Processing-Effekte oder speziell angepasste Shader Libraries?

David Helgason Wir konzentrieren uns nicht speziell darauf. Aber viele in Unity 5 enthaltene Features werden für jeden, der optisch fesselnde Inhalte produzieren will, von großem Nutzen sein. Es gibt auch eine Menge Updates für unser Animationsframework, um beispielsweise das Timing von Events einfacher zu machen oder verschiedene Abläufe zu verknüpfen, die bisher getrennt waren. Wir versuchen zunächst den Bedürfnissen der Spieleentwickler nachzukommen, aber vieles dieser Technologie kann auch in anderen Bereichen angewendet werden.

Making Games Habt ihr bei der Entwicklung eigentlich auch die Filmbranche im Hinterkopf? Für Crytek ist sie ja beispielsweise ein nicht ganz unwichtiger Teil des Geschäfts.

David Helgason Na ja, die Filmbranche ist natürlich ein Teil des Geschäfts, da eine Reihe Studios Unity zum Beispiel für Werbespots nutzt. Aber das ist für uns kein entscheidender Faktor.

Making Games Die Unreal Engine und CryEngine folgen in Bezug auf das Geschäftsmodell nun in gewisser Weise eurem Beispiel. Was hältst du persönlich davon?

David Helgason Ich finde, dass es nicht das Gleiche ist. Meiner Meinung nach denken die Nutzer unterschiedlich darüber. Die fünf Prozent Gewinnbeteiligung von Epic für die Unreal Engine 4 klingen nach wenig, aber es summiert sich. Das größere Problem ist, dass viele Unternehmen gerade einmal 10 bis 15 Prozent Gewinn machen. Das heißt, dass diese Firmen noch die Hälfte oder ein Drittel davon abtreten müssen.

Making Games Euer Geschäftsmodell wird sich in Zukunft also nicht in diese Richtung entwickeln?

David Helgason Wir denken, wie gesagt, immer wieder darüber nach, wie wir unser Modell verändern könnten. Im Lauf der Jahre haben wir einen immer größeren Teil unseres Angebots kostenlos gemacht und ich hoffe, dass wir das in Zukunft beibehalten können. Aber anstatt uns darauf zu konzentrieren, wie wir über Beteiligungen oder Ähnliches mehr Geld einnehmen können, legen wir unseren Fokus in letzter Zeit immer mehr darauf, wie wir Entwicklern mit Unity zum Erfolg verhelfen können.

Für Entwickler gibt es zwei überlebenswichtige Themen: einerseits ein gutes Spiel kosteneffizient zu produzieren und andererseits die Spieler dafür zu begeistern. Daher haben wir zwei Initiativen gestartet. Die erste ist das Unity Cloud Cross-Promotion Network, das gerade über die Betaphase hinaus ist. Wir haben erst vor kurzem einen starken Traffic-Anstieg im Netzwerk verzeichnet. Es wird also langsam, aber sicher, wirklich relevant für die Entwickler, um für ihre Spiele zusätzlichen Traffic zu generieren. Unsere zweite Initiative ist die Übernahme der Videostreaming-Lösung Everyplay. Es ist kostenlos, jeder Entwickler kann es also in sein Spiel integrieren. Und die User können dann ihre besten Spielmomente einfach per Knopfdruck teilen. Die Entwickler bekommen dadurch viral erzeugten Traffic, weil sich andere Spieler die geteilten Videos ansehen. Den Dienst gibt es erst seit einem guten Jahr, aber wir haben schon von vielen Entwicklern gehört, dass Everyplay ihr wichtigstes Werkzeug für viral erzeugten Traffic ist. Jede zweieinhalb Sekunden registriert sich ein neuer User, das macht es sehr leistungsstark. Damit wollen wir also sicherstellen, dass unsere Kunden mehr Erfolg haben.

Making Games Ein möglicher nächster Schritt wäre für euch, Spiele zu verkaufen, ihr habt ja eine riesige Bibliothek. Der Mobile-Markt ist stark umkämpft, wäre das für euch interessant? Vielleicht etwas Ähnliches wie Steam aufzubauen, aber speziell für Unity-Spiele?

David Helgason Ich denke nicht, dass das etwas für uns wäre. Wir wollen lieber den Traffic und dadurch den Umsatz der Spiele steigern. Teil der Übernahme von Everyplay war auch das Werbenetzwerk »GameAds«. Die Entwickler können durch den Austausch von Downloads den Wert ihres Netzwerks steigern - sie verlieren den User ja nicht, weil sie ihm ein anderes Spiel empfehlen. Der Wert wird quasi aus dem Nichts generiert. Oder, um ein anderes Beispiel zu nennen: Wenn du selbst viel Traffic, aber eine schlechte Monetarisierung hast, kannst du den Traffic auch an jemand anderen verkaufen, bei dem es genau umgekehrt läuft.

Making Games Ihr versucht also genau genommen, innerhalb eurer Entwicklercommunity den Austausch zu fördern?

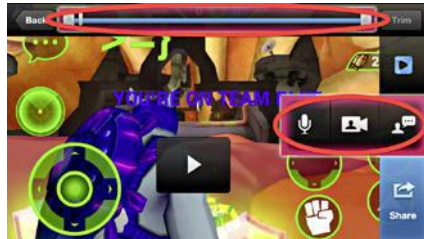


»Max: The Curse of Brotherhood« ist eines der ersten für die Xbox One erschienenen Unity-Spiele.



Sonys VR-Headset »Morpheus« wird auch für Unity ein Thema sein: »Unsere Partnerschaft mit Sony umfasst alle Geräte.«

 »Wir legen unseren Fokus immer mehr darauf, wie wir Entwicklern mit Unity zum Erfolg verhelfen können.«



Neben dem Unity Cloud Cross-Promotion Network ist die Everyplay-Akquise eine weitere Initiative von Unity, den Entwicklern bei der Traffic-Steigerung zu helfen.

»Eines der zentralen Merkmale von Unity ist die Multiplattform-tauglichkeit.«

David Helgason Genau. Das alles läuft im Moment langsam an und funktioniert bisher wunderbar. Im großen Kontext wird das Bild klarer: Rund zwei Milliarden Menschen sind mit App Stores verbunden. Innerhalb dieser riesigen Menge findest du viele kleine Publikumsgruppen, die sich zum Beispiel für bestimmte Genres oder Titel begeistern. Aber auch diese Gruppen können immer noch aus Millionen von Usern bestehen. Mit Diensten wie Everyplay können wir den Entwicklern helfen, direkt mit ihrem Publikum in Kontakt zu treten. Das wird immer wichtiger, denn aktuell gibt es abgesehen davon eigentlich nur noch eine andere Strategie: hoffen, dass man es in die Top 10 schafft, oder die User kaufen, was aber schon wieder Budget voraussetzt. Oder einen hohen Durchschnittserlös pro User, den die meisten Spiele einfach nicht generieren; zumindest nicht in einer Größenordnung, dass die Entwickler damit teure Werbung schalten können.

Making Games Hältst du Pay-per-Play noch für ein rentables Geschäftsmodell? Fireproof Games beweist ja momentan gerade mit »The Room 2«, dass es funktionieren kann.

David Helgason In den letzten Jahren konnten wir auf der GDC Monokulturen beobachten. Im einen Jahr musste jeder Facebook-Spiele haben, im nächsten drehte sich alles nur um Mobile, und noch ein Jahr später hieß der Trend Free2Play. Jetzt kann man beobachten, dass sich die Entwickler wieder auf unterschiedliche Modelle verteilen, zum Beispiel über Premium-Angebote oder DLCs. Du siehst auch Entwickler, die von Mobile zurück auf stationäre Konsolen oder den PC wechseln. Das liegt meiner Meinung nach daran, dass sie die Kommunikation mit den angesprochenen Publikumsgruppen gelernt haben. Dadurch bekommen wir wieder thematisch spezifischere Spiele und weniger Titel, die darauf ausgelegt sind, die vermeintlich meisten Geschmäcker irgendwie zu bedienen.

Making Games Cross-Plattform ist für zahlreiche Entwickler noch immer ein großes Thema. Wie wollt ihr diesen Trend unterstützen?

David Helgason Dafür ist Unity sehr gut geeignet. Du schreibst deinen Code einmal und

kannst dich dann auf die Feinheiten konzentrieren. Natürlich musst du dich immer noch um die Steuerung kümmern und sie für das jeweilige Gerät anpassen. Oder die unterschiedlichen Bildschirmgrößen. Aber wir möchten, dass sich die Entwickler auf diese Dinge konzentrieren können und sich keine Sorgen um Abstürze oder das Rendering machen müssen. Eines der zentralen Merkmale von Unity ist die Multiplattformtauglichkeit. Das sieht man daran, wie schnell die Entwickler es schaffen, ihre Spiele von einer Plattform auf die andere zu portieren.

Making Games Das betrifft vor allem Free2Play-Titel und MMOs, oder? Die Entwickler dieser Spiele tun sich schwerer.

David Helgason Stimmt. Wenn man die Technologie einer bestimmten Plattform besonders in Anspruch nimmt, wird die Portierung schwieriger. Aber trotzdem haben wir auch in diesem Bereich schon fantastische Umsetzungen gesehen.

Making Games EA ist einer eurer größten Kunden, nutzt Unity aber vor allem, um damit schnell Prototypen zu bauen, und wechselt dann in die hauseigenen Engines, wenn alles funktioniert. Ist das für euch in Ordnung oder stört euch das hin und wieder?

David Helgason Ich kann zu EA im Speziellen nichts sagen, aber wir sehen immer wieder, dass Firmen Unity für ihre Prototypen verwenden. Doch sie nutzen unsere Engine auch immer häufiger, um damit dann das komplette Spiel zu bauen. In den nächsten Jahren werden wir einige ziemlich spektakuläre Spiele sehen.

Making Games Kannst du drei oder vier Unity-Spiele nennen, die dich beeindruckt haben und die zeigen, was mit der Engine alles möglich ist?

David Helgason Ich habe in letzter Zeit häufig »Dead Trigger 2« gespielt. Auch »Device 6« ist ein unglaubliches Spiel. Dann wäre da noch »Battle Bears«, das von einem wirklich kleinen Team gemacht wird. Ich finde zudem »Kerbal Space Program« einfach nur fantastisch. Und nicht zu vergessen »Rust«, das sich momentan wie ein Lauffeuer ausbreitet.

Interviews: Heiko Klinge



»Kerbal Space Program« (links) und »Rust« (rechts) zeigen, was mit Unity alles möglich ist.



YOUR KEY TO THE GERMAN GAMES INDUSTRY

MAKING GAMES MAGAZINE: 10,000 READERS
CENTRAL EUROPE'S MOST RELEVANT MAGAZINE FOR GAME DEVELOPERS

MAKINGGAMES.DE: 8,000 UNIQUE VISITORS
GERMANY'S BIGGEST WEBSITE ABOUT GAME DEVELOPMENT

FACEBOOK.COM/MAKINGGAMES: 5,200 FANS
EUROPE'S LARGEST GAME DEVELOPER COMMUNITY ON FACEBOOK

MAKING GAMES MAIL: EMAIL DATABASE
WITH MORE THAN 9,000 B2B-CONTACTS

MAKING GAMES PROFESSIONALS: LEAD DATABASE
WITH MORE THAN 700 FULLY QUALIFIED GAMES PROFESSIONALS

MAKING GAMES TALENTS: GERMANY'S MOST SUCCESSFUL
RECRUITING EVENT FOR THE GAMES INDUSTRY

KEY PLAYERS: THE WORLD'S BIGGEST GAMES INDUSTRY
COMPENDIUM WITH MORE THAN 80 COMPANY PORTRAITS

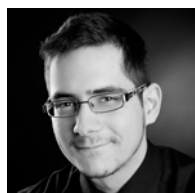
TALK TO US!

PROJECTS@MAKINGGAMES.DE



DECISION-MAKING IN COSMONAUTICA

Achieving a natural behavior of NPCs is a great challenge but also necessary for immersion. Philipp Erler explains the artificial intelligence in Cosmonautica.



Philipp Erler
just finished his studies of
Software Engineering at the
University of Applied Sciences
in Esslingen.

Philipp was a programmer at Chasing Carrots while writing his bachelor's thesis »Decision-Making in a Group of Artificial Intelligences in Different Simulated Environments«. He worked on the prototype of a racing game before moving on to work on »Cosmonautica«. More info on the game at www.cosmo-nautica.com.

The bachelor's thesis and the behavior tree core system:
www.pplusplus.lima-city.de/bachelorsthesis.html

Cosmonautica is the second game of the small indie studio Chasing Carrots KG in Stuttgart, Germany. The game combines »Sims«-like management of the crew members in the player's spaceship with trading and space fights.

Instead of getting direct orders from the player, the NPC crew members choose their actions on their own within certain limits. The AI system must therefore make these decisions. Since the crew members' behaviors are a main part of the game and crucial for the player's success, the decision-making system must support quite complex behaviors. In short, it is meant to choose the work which the crew members do or how they satisfy their needs. Then, the system should control how they do it to ensure that everything looks natural.

Cosmonautica was in an early stage of development when the work on the decision-making system started. Some behavior related systems, like the need system for example, did already exist; but the behaviors were just placeholders. Therefore, the new decision-making system had to fit between these systems. The decision-making core system is

meant to be a part of the code base Chasing Carrots will use for future games.

Behavior Trees

State machines are very simple but can hardly be re-used. Fuzzy logic or Markov systems as extension for state machines or stand-alone are not necessary for Cosmonautica because real probabilistic behaviors and multiple states are not desired. Goal-oriented action planning is a great technique for action planning, which is not required. But utility functions, a part of the goal-oriented behaviors, made their way into Cosmonautica. Behavior trees combine simplicity with good behaviors and awesome flexibility.

Tree Structure

Behavior trees are directed, acyclic graphs made of nodes and edges. To be more exact, behavior trees are rooted and directed trees. Nodes need to know only their child nodes, which leads to a hierarchy. All nodes share the same interface, making it easy to exchange them, which in turn is the reason for the flexibility of the behavior trees. A behavior tree can be started very simple as a placeholder



Cosmonautica combines »Sims«-like crew management with trading and space fights.



What should Calvin (lower left-hand corner) do? Moreover, where? Should he work or satisfy his needs? How long and how often? These are difficult questions for humans and even more so for a computer.

and can be easily extended later on with working states in between.

Node States

Each node sets its own state every time it is evaluated, then it returns this state to its parent node. Alex J. Champandard says in his talk »Understanding the Second Generation of Behavior Trees« that he tried many possible state combinations but the following are the best:

- **Invalid** – This state is set in the constructor of the base node class. Therefore, the invalid state indicates that the node's evaluation method has never been called. It can also be set to show that an exception occurred.
- **Successful** – This state is set when a node evaluated successfully. For example, a node that teleports the player character to another place could use this state to show that the player has arrived.
- **Failure** – Failure is the opposite. Using the previous example, this state could be set if the teleport fails because the player lacks the resources.

- **Running** – The running state is a bit special as it may persist through several evaluations of the node. In the teleport example, this state would show that the teleport takes time to open a portal. Once the portal is opened, the teleport node would set its state to successful.
- **Aborted** – Any node may cause the teleport node to be reset. Then, the aborted state is used. This could happen when the character received damage while opening the portal.

	State Machines	Fuzzy Logic / Markov Systems	Goal-Oriented Behavior	Rule-Based Systems	Behavior Trees
Simplicity	+	-	o	-	+
Separation of Game Design and Programming	+	o	o	+	+
Flexibility	-	o	o	o	+
Behavior Quality	-	o	o	+	+
Efficiency	+	-	-	+	o
Conclusion	Very simple	Probabilistic	Great for planning	Limits not reached yet	High flexibility

This overview compares different decision-making techniques and their characteristics. Note that behavior trees might not be the best choice for your own decision-making system. As Millington and Funge say in »Artificial Intelligence for Games«, it can be cumbersome to achieve some behaviors with behavior trees. This applies especially to state-like behaviors.

The crew members' behavior is a huge part of the game, so the decision-making has to feel natural to the player.



There are two events reacting on the states of a node. The »on initialize«-event is triggered before the evaluation but only if the node's state is not running. The »on terminate«-event is triggered after the evaluation but before returning its state, only if it is not running. Both events may execute custom code. In the teleport example, the on-initialize-event could start an animation, the node evaluation would check if the teleport preparation is finished, and the on-terminate-event would set the caster's new position and stop the animation.

During the development of Cosmonautica, no situation occurred that required different states. In fact, the whole behavior tree system applied by Chasing Carrots is based heavily on the behavior tree starter kit by AiGameDev.com.

Basic Node Types

Millington and Funge suggest the following behavior tree node types:

- **Leaf Nodes** – They are at the end of the behavior tree and have no child nodes.
 - **Action** – These nodes alter the state of the game or a game object.
 - **Condition** – They check a fact in the game.
- **Composite Nodes** – These nodes can have multiple child nodes. They base their own evaluation on the return values of their children.
 - **Sequence** – These nodes try to evaluate their child nodes until one returns something other than successful.
 - **Selector** – They try to evaluate their child nodes until one returns other than failure.

- **Parallel** – These nodes try to evaluate their child nodes until a specific number of successful or failure is returned. In contrast to the other composite nodes, parallel nodes do not commence their evaluation at the last running node. Instead, they start always with the first child.

- **Decorators** – These nodes have only one child node. They change the way this child is evaluated. Many different types of decorators are imaginable. For example, they can change the child's return value or act as a breakpoint.

Behavior Tree Evaluation

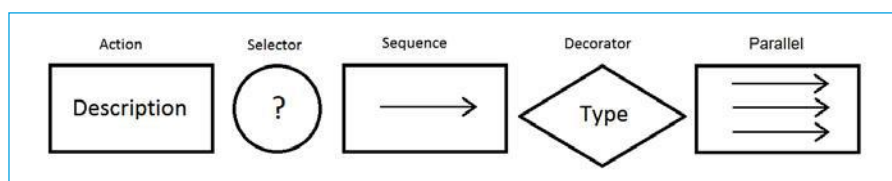
Behavior trees are evaluated from the root every time the tree is updated, which can happen with a significantly lower frequency than the normal frame rate. In Cosmonautica, the update of the crew members' behavior trees is done every ten in-game minutes, which equals about two real seconds. This relieves the CPU quite a lot but leads to a more difficult optical representation of the crew members.

Each node of the behavior tree tries to run through all of its child nodes. The behavior tree can be designed in such a way that the root node's result indicates if the tree found a fitting behavior for the situation of the object it currently controls.

Implementation

Once the behavior tree has been assembled, there are usually no more changes as long as the game runs. Therefore, a vector of pointers to the child nodes is the best way to achieve the special tree structure. The only exceptions are decorators, which only have a single child node and leaf nodes without any children.

Whether the crew member should work is controlled inside the crew member class through his assigned tasks and the game time. This class has a small internal state machine for the walking and doing state.



The most important node types of the behavior tree notation used by Millington and Funge.

Finding the Best Activity through Utility Functions

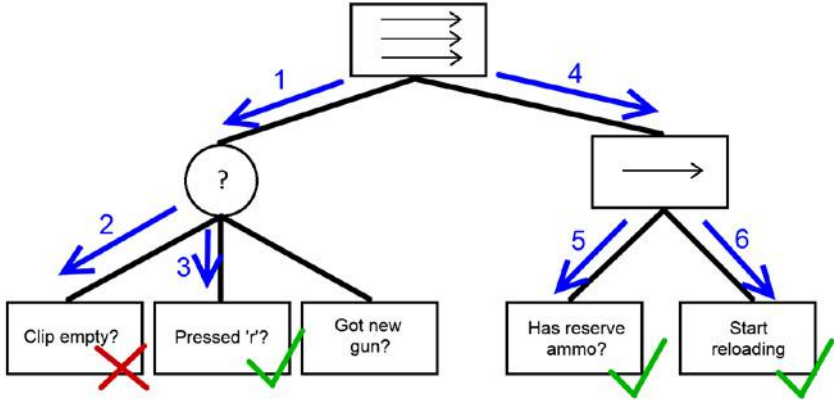
There are several reasons why the actual search for the best activity is not modeled into the behavior tree but is done within action nodes. The first reason is that the activities in Cosmonautica are added by placing rooms and they are removed when the providing room is sold or destroyed. To model this into the behaviour tree would require a regular modification of the tree while the game runs, which might have nasty side effects. If all approximately hundred rooms of a big ship were placed, the behavior tree would then grow by at least each two hundred tasks and activities plus many management nodes. This would be simply too much to keep a good performance. Another reason why the activity search is not modeled in the behavior tree is that not all possible activities are already known and the activities are planned to be changed by mods. Therefore, maintaining the behavior tree will be almost impossible. With the approach of putting the searches into action nodes, only utility functions have to be added for every new type of activity or task.

In order to find the current need, all needs of a crew member are considered beginning with the most urgent. A need can only be the current need if an activity exists, which can satisfy this need. The search for the best activity has the same conditions as the search for the current need. Therefore, both searches

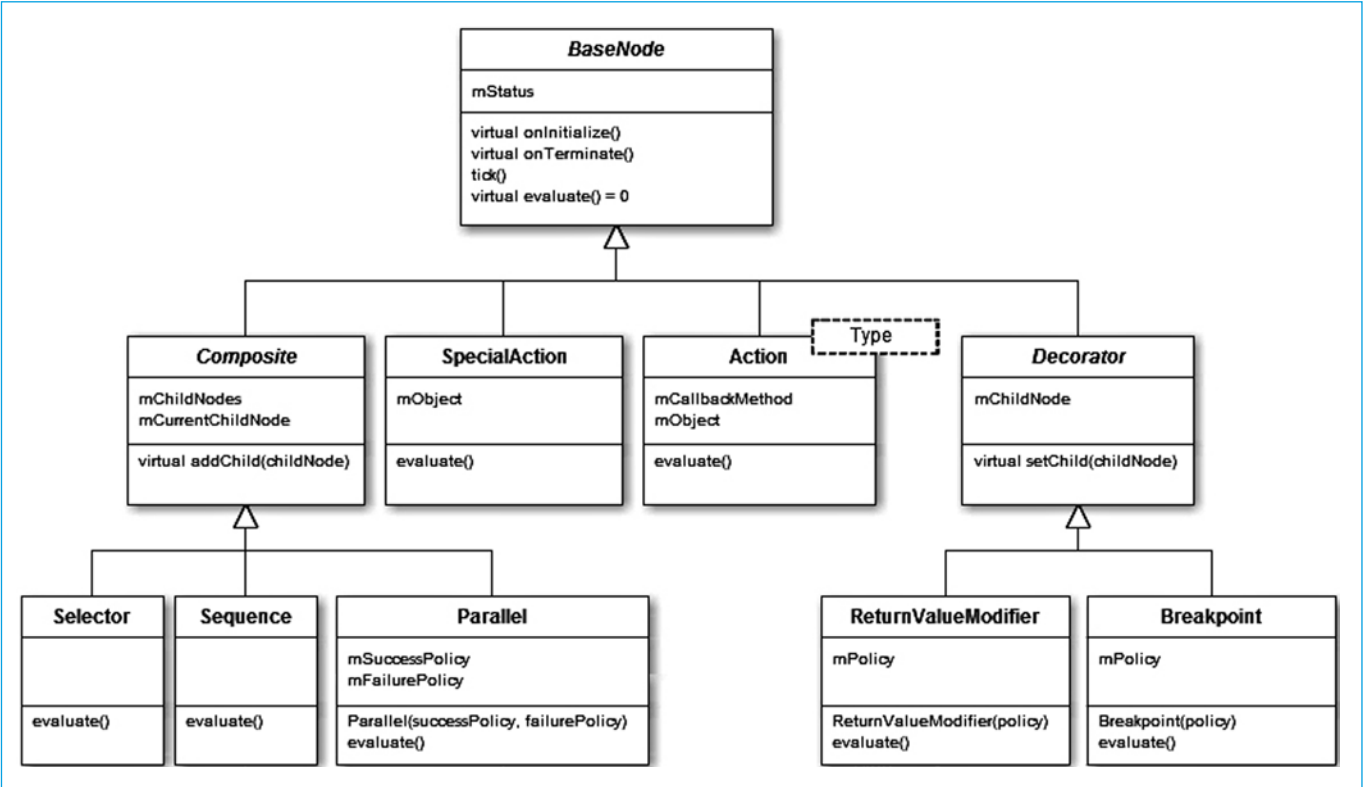
Example: Evaluation of a Behavior Tree

The parallel root node runs its first child, a selector, which looks for a reason why the gun should be reloaded. This node looks for a »successful« child node and returns successful if one is found. Now, the root node runs its second child, a sequence node, which checks

if the player has reserve ammo. If this check is successful, the sequence node starts the reloading. The parallel node here has the policies »all for successful« and »one for failure«, which means that its second child is not evaluated if the first child returns »failure«.



are pulled together to decrease the calculation time. To find the best activity, all activities lowering the current need's urgency are looked at. Each of these activities gets its utility value calculated. The calculation itself is done with the utility-system technique from the category of the goal-oriented behavior.



All node classes are derived from the abstract base node. This class manages that the »initialize« and »terminate« events are triggered, calls the evaluation method, sets its result as the current node state and returns it to the caller. The »Composite« class manages the vector of its child nodes. The »SpecialAction« class enables custom »initialize« and »terminate« events and provides special member variables.

The AI system makes sure that the crew members work or satisfy their needs when it makes sense. The crew members also choose the room for these activities with some intelligence.



In the case of the crew member activities, the utility is calculated out of this data:

- The **distance** between the crew member and the possible activity
- The **effect** of the activity on the current need as well as the effect on other needs
- The **urgency** value of the current need
- Whether the possible activity is the **current** activity or the target activity
- The length of the **queue** in front of the activity
- How much **dirt** and **damage** the activity does to the room of the activity

Each of these values has a factor assigned, which defines how much the value influences the overall utility of the activity. For example, the distance between the crew member and the possible activity is mostly meant to decide between equal activities in different rooms. Therefore, its factor is balanced to decide for a less effective activity over a more effective only if the latter is at the other end of the ship. The walk speed or the required time to satisfy a need by doing the activity are not considered anymore because it led to a hard-to-control nonlinear change of the utility value over time.

If several activities have exactly the same utility value, one is chosen randomly by the behavior tree to improve the diversity. In the case of the fitness room, this makes the crew members choose randomly between lifting weights and running on the treadmill. Both activities have the same effects on the character, but are animated differently.

Prioritizing Tasks

Finding the best task is a little different. Tasks have a priority value stored instead of a utility value. This priority is calculated in a similar way but it is independent from a certain crew member. It depends only on the room providing the task. The priority values are calculated differently for each task type. The priority of the repair task for example is based on the

damage of the providing room and its importance. This importance value causes crucial rooms like the engine room to be repaired more often than e.g. cargo rooms. Other tasks, such as the »man turret« tasks, have simply the maximum priority value of one but only if the ship is in a space fight. However, each task type needs a separate priority calculation.

Space Station Activities

The search for an activity on a space station is another case. Crew members find their current need just like on the ship by choosing the most urgent need which can be satisfied. Because space stations have activities for all needs, the current need is simply the most urgent one. Afterwards, crew members choose a random activity, which satisfies the current need. Therefore, only the type of the activity and the need's urgency are considered for the space station activities. Choosing the activities randomly improves the diversity of the behaviors a lot. The behaviors and decisions on the space stations can be much simpler because all the information the players get about it is a short text in the GUI.

Utility Functions

Our experience shows that the utility values themselves are not important at all. Only their relation to each other is relevant for ordering the activities. Utility systems usually need to compare all activities with every need what leads to »O(needs*activities)« in time. By choosing the most urgent need before searching an activity, the complexity is decreased to »O(needs+activities)«.

Conclusion

The AI system makes sure that the crew members work or satisfy their needs when it makes sense. The crew members also choose the room for these activities with intelligence. They wait in a queue for a reasonable time and they try to do their assigned tasks as good as they can.

Bibliography

AIGameDev.com (Champandard, Alex J.)

»Behavior Tree Starter Kit«

<https://github.com/aigamedev/btsk>

»Understanding the Second-Generation of Behavior Trees - #AltDevConf«

<http://aigamedev.com/insider/tutorial/second-generation-bt/>

Millington, Ian and Funge, John

»Artificial Intelligence for Games, 2nd Edition«

Boca Raton: CRC Press, 2009.

The behavior trees are up and running in Cosmonautica and they are controlling the crew members. The utility functions have proven to be useful for situations where behavior trees would have become too big, for example the search for an activity.

Behavior trees have shown these advantages:

- The **flexibility** is very high as shown through the usage of behavior trees for the AI ships in space fights, which was not planned at the beginning.
- The **maintaining** of existing behavior trees is fast and easy even without a graphical modeling tool.
- The **code** of the BT nodes can easily be re-used as well as parts of the behavior trees.
- The **quality** of the behaviors is high enough to support everything we need.

Nothing is perfect. Behavior trees are no exception and showed these disadvantages:

- The **distribution of complexity**, especially between the behavior tree itself and its action nodes' code is tricky. It can be useful to combine or split behavior trees. Putting too much complexity into the action nodes makes the behavior tree harder to understand and decreases its re-usability. Usually, the entire complexity cannot be modeled into the behavior tree and trying it will result in a gigantic and slow behavior tree.
- Splitting up similar things, like checking all activities and then choosing one, might be required to re-use nodes but causes **code duplication** and **inefficiency**.
- **Debugging** a behavior tree with the usual tools is difficult, mostly because of the useless call-stack. It is very helpful to see every node's state of the entire behavior tree with one glance. Therefore, a debug print or an external tool is recommended.

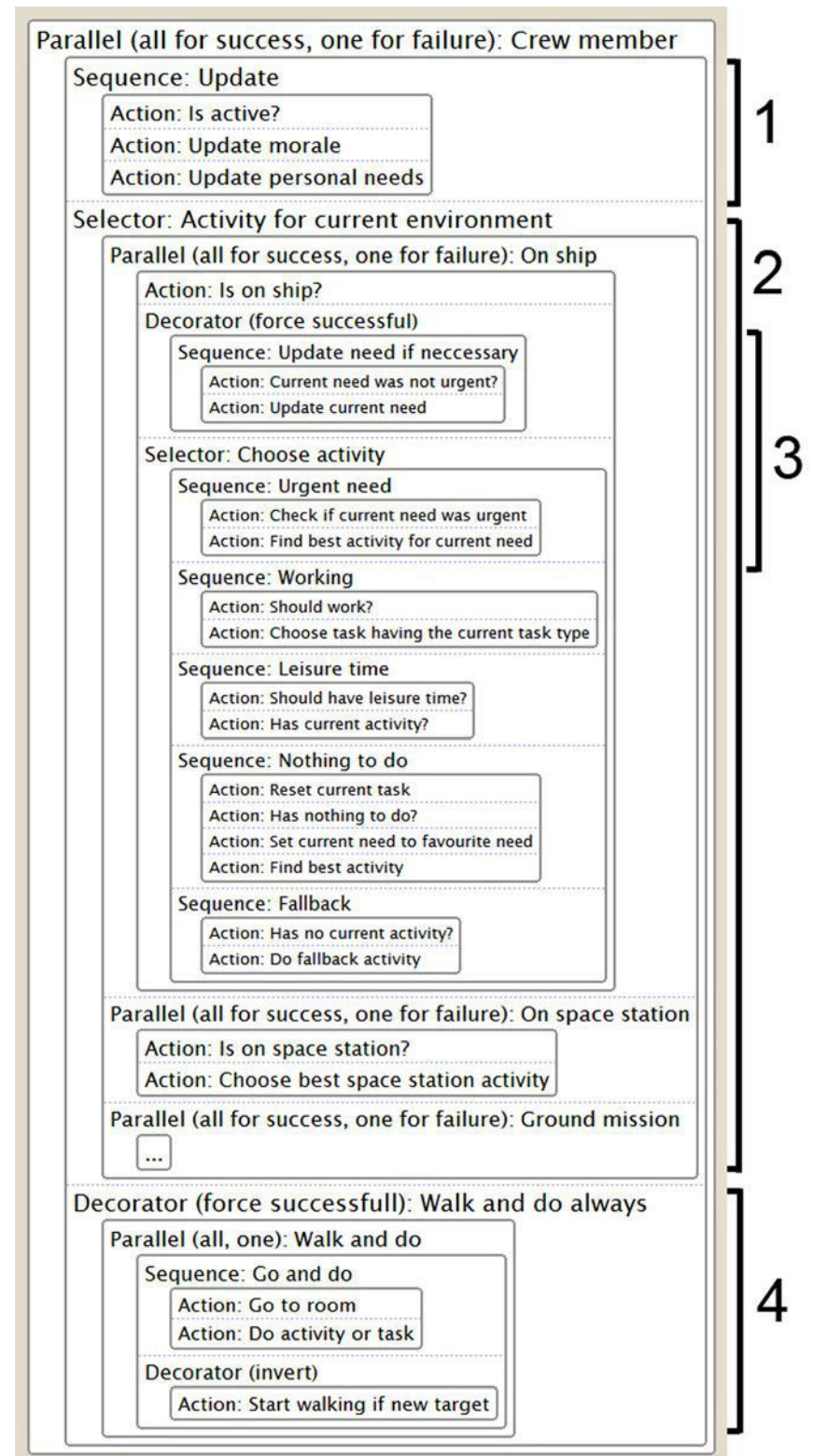
We gained valuable experiences through Cosmonautica. We learned that utility functions are great as action nodes where a normal node structure is not applicable. We also came to understand how powerful behavior trees are. On the other hand, we noticed that more and more advanced behaviors can also harm the game. Crew members in Cosmonautica for example do not avoid each other anymore during activities. Instead, they wait in a queue in front of the activity giving the player a chance to see the bottlenecks in the ship. And we learned that state-like behaviors in behavior trees usually require a small state machine controlled by action nodes. This decreases the complexity of these state machines to the absolute minimum and therefore increases their re-usability. But it can also make the entire behavior tree harder to understand. However, this is still much better than just a single big state machine.

Philipp Erler

The Crew members' Behavior Tree

The crew members' behavior tree consists of several parts:

1. Update the morale and all needs if the crew member is alive.
2. Find the best activity for the current environment.
3. Stick with the current need if it is urgent or was urgent since the current activity started.
4. Update walking or do the activity if the crew member has arrived. Then start walking to the new activity if the current activity has changed.





DEVELOPING PLAYERS

WHAT CASUAL GAMES CAN LEARN FROM MMOS

Casual games have in common with MMOs that their players evolve. Richard Bartle explains how casual games have to adapt to changing tastes while still remaining accessible to new players.



Richard A. Bartle
is Honorary Professor
at the University of Essex.

Richard is one of the pioneers of the MMO industry. He has a Ph.D. in Artificial Intelligence from the University of Essex where he has also co-written the first virtual world »MUD« with Roy Trubshaw in 1978. His Player Types model has been widely adopted by the MMO industry since he first introduced the idea in 1996. In 2000, Richard headed the online games division of the start-up Gameplay and went on to consultancy after that. He returned to lecturing at the University of Essex in 2002 and is today teaching part-time at the School of Computer Science and Electronic Engineering of Essex University.

Some time ago, I went to a talk on Artificial Intelligence and Games, in which one of the speakers began: »I know nothing about games. I don't play them and I don't like them. However, I do know about Artificial Intelligence, and I'm told that some of this stuff may be of interest to you games people«. Needless to say the gamers in the audience weren't impressed. I am about to commit a similar indiscretion.

My background is in MMOs. Although I also know a little about casual games, a lot of people probably know a lot more about them. But still, some of my knowledge might also be interesting in regard to casual games. MMOs have a very long history in computer game terms, with almost all of them descending from a single virtual world called »MUD« that was developed in 1978. Some of the things we're seeing now with casual games, we've seen before – sometimes

several times before – with MMOs. Long-term MMO designers and developers recognize many of the problems (or, handled properly, opportunities) that the casual games industry can expect, and can tell you how to address them. The history of MMOs is something I do know about. I co-wrote »MUD« (Figure 1).

Accessibility for New Players

Social, casual and mobile games are great at engaging brand-new players. People who weren't previously much into playing games can pick these up and engage with them very quickly. My wife insists that she doesn't play »Bejeweled« (Figure 2) much, but the grid of her fingerprints all over our iPad tells a different story. Casual games are far more accessible than traditional videogames, so it's not surprising that they are far more popular. There are enormous amounts of money to be made here! However, people who play games don't just

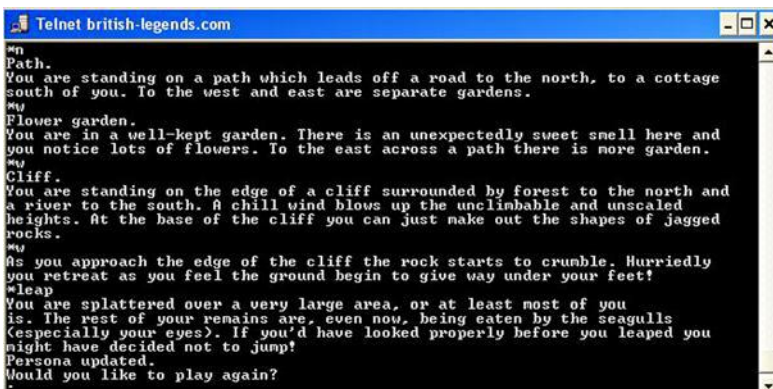


Figure 1 »MUD« was the first Multi-User-Dungeon and is the oldest virtual world in existence.



Figure 2 Casual games like »Bejeweled« are a lot more accessible than traditional videogames for new players, so their widespread popularity comes as no surprise.

play them – they absorb them. They come to understand what they're playing. Inevitably, after they have internalized the workings of a particular game, they'll want to progress to more challenging content. When children are small, they like to read »The Very Hungry Caterpillar«. When they grow up, they don't want to read anything remotely to do with hungry caterpillars, they've moved beyond that. Their tastes in books have developed, just as their tastes in music, film and television have developed. The same applies to games.

The games that people play today are training them in the ways of games as a medium. The games that those people play tomorrow will have to be more advanced. People are like machines that are programmed to learn. Once they have learned a game, its attraction diminishes.

Tastes Change

This has been known about in MMOs for decades – since the early days of »MUD«, in fact. Players' tastes change. What's more, they change in predictable ways.

»MUD« was the progenitor of virtual worlds. By definition, when new people started playing it, they hadn't come across anything like it before. The first thing they would try to do when they stepped into the virtual world would be to try to kill a creature or the other players (well, their characters). After a while, they would move on to exploring the world and its functionality. Following that, they would try to win the game in its own terms (which meant racking up points to reach the highest level). Eventually, they would end up as grizzled old-timers who spent their time pottering around in the game world mainly so as to hang about with the other veterans there that they knew. In Player Type Theory terms, new players began as killers, then became explorers, then became achievers, then became socializers (Figure 3).

We call this the main sequence of MMO player development. There are actually three more sequences, but those require a third dimension on the Player Types graph to explain. For example, some players don't start off killing one another, they just try everything physically possible. Can they walk through walls? Can they climb trees? What happens if they jump off a cliff? Ooh, a bird, can I catch it? After that, they start exploring and achieving as normal, but instead of ending up as socializers they become more like gurus who understand the game implicitly and play it to reveal meaning.

All the development paths that players follow have the same general pattern. Players begin by trying to establish the boundaries of the world, socially as well as physically (Figure 4). Then they explore within those boundaries, they try to succeed on the MMO's terms based on what they've learned, then they relax on their own terms. This happens to all MMO players who play for any length of time. Casual games aren't MMOs, but a similar thing happens.



Figure 4 When people start playing MMOs like »The Elder Scrolls Online« most of them usually start by trying to kill either other players or an enemy provided by the environment to establish the boundaries of the game's world.

Players find what games are on offer, then explore which of these they like; they get good at the ones they play, then they move on.

Importantly, this affects their relationship not only to individual games, but to all games that are similar. A grown-up wouldn't be interested in »The Very Thirsty Caterpillar« or »The Very Hungry Silkworm« unless it was a parody.

This happens in MMOs and also in other games. For an explanation of why it happens, read Raph Koster's book, »A Theory of Fun for Game Design«. If you're a designer, you should already have it. If you don't, buy it – it only takes a couple of hours to read and is worth the effort.

Now although all game players progress relentlessly through increasingly challenging content, they don't all progress at the same rate. Every player has their own pace. With MMOs, because players tend to play the same one for quite a while, we can see this as it happens.

Figure 5 shows the characteristic shape of a graph showing how many players there are at different levels a few weeks from launch. There will be more players at the lowest levels, because some players will try the game but not like it so leave. There will also be more players at the highest levels, because very experienced players will descend on the game and munch their way through it at a tremendous speed – they're called content locusts for this reason. Most players, though, will be enjoying their time playing and feel no need to rush. They'll be progressing steadily, and will be pretty well evenly distributed between the two extremes.

Figure 6 shows what the same graph for the same MMO would look like after several months. Players may progress at different rates, but they do progress. Eventually, they reach »the end«. When they reach the end, they expect end-game content. This is what casual games players will also want.

At the moment, there are millions of people playing casual games. In so doing, they're learning to play games. Over time, we'll see the same curve as for MMOs. There'll be a spike at the start for people who didn't take to games as children and didn't get on with them when they tried them as adults (like the AI professor I mentioned earlier); there'll be a spike at the end

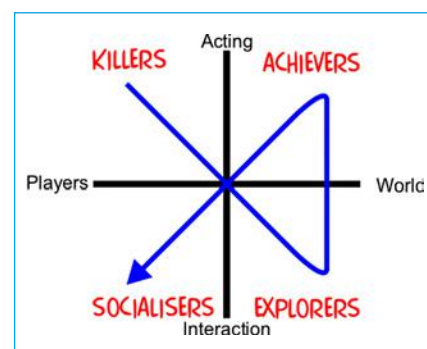


Figure 3 According to the Player Types model, players go through different stages as they learn the dynamics of a game.

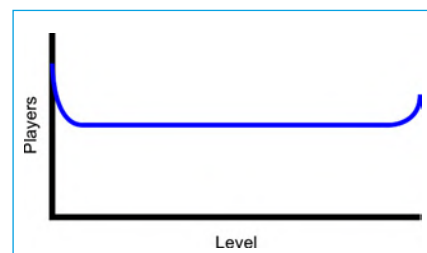


Figure 5 A few weeks after the launch of a game, more players will be bunched at the lowest and highest levels because they're new to the genre or have a lot of experience. Most players are evenly distributed between the extremes because they progress steadily.

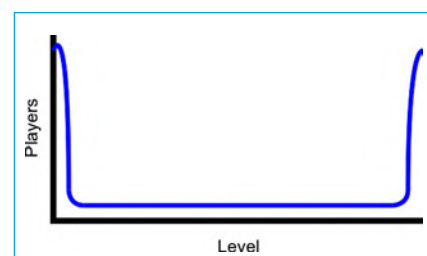


Figure 6 Even players that progress steadily reach the end, turning the even distribution into a spike at the higher levels.

for people who want more sophisticated content than what they went through to get there. Although you will still have people playing your current games in future, they won't be the same people and there won't be so many of them.

Players Move On

So, players will move on. This raises the question: where will they move? The answer from MMOs is that they'll tend to move where you point them! So long as what you offer is indeed progress in terms of sophistication, they'll usually be happy. Knowing this, what can you do?

Surprisingly, many developers answer this the wrong way. They know that left to their own devices, players who want more sophisticated content will try more sophisticated games. So they graft casual features onto more sophisticated games in the hope of luring casual players to them early. This can work, but it's not very effective. Adding illustrations to »War and Peace« isn't going to attract many young readers to it.

The better solution is to graft more sophisticated features onto casual games. That way, you take the audience with you. Fiction for older children still has illustrations, but it has much longer stretches of text. Players want to progress at their own pace, remember – not by making a leap of faith into what looks like a trap.

Higher Level of Sophistication

How do you make a game more sophisticated? You either add more detail or more abstraction. In the long term, abstraction works better than detail. For example, when hidden object games started out, you would see a scene with perhaps 20 objects on it. Nowadays, you look at a scene of random stuff washed up on a riverbank under a bridge and it's more like hunt-the-pixel. There could be a couple of hundred items there. If you're into hidden object games, you'll have no trouble; if you're not, you'll be staring fruitlessly until the hint option kicks in. With games that add detail, what happens is that eventually players burn out. They've drilled into the genre so much that it has little to offer them. They'll look for something else to drill into – perhaps not even a game. Somebody must be buying those 6,000-piece jigsaw puzzles ...

Making games more abstract (in the sense of encouraging higher-level thought processes) is a better approach for two reasons. Firstly, there are more people who like to expand their skills than there are who like to sharpen them. Secondly, it's easier to take your players with you this way. You can address a yearning for ever-harder strategic reasoning by a change of subject; you can't address a yearning for ever-harder detailed analysis the same way.

It's important to note that although I'm suggesting that the players of the future will want more sophisticated content, that doesn't mean there's no place for the older-style games with less sophisticated content. If there's not a gateway game and a graduated series of intermedi-

ate games, people won't have a way to reach the advanced content. Most players will no longer play the simpler games, but to play the sophisticated games they need to have played them.

Too esoteric to survive

I'm going to end with a mention of a game genre that was spun from the same cloth as early MMOs, but that died a slow death. This is the adventure game. Many of the youngsters reading this won't have heard of adventure games, but in the mid-1980s they dominated the PC games industry. They successfully made the transition from text to graphics, but then they all but disappeared from the market for nearly a decade. This is because they became so esoteric that they lost all but their most hardcore audience. No-one could fathom the puzzles.

Here's an example, from »Gabriel Knight 3: Blood of the Sacred, Blood of the Damned« – one of the last and most critically acclaimed of all adventure games (Figure 7).

The Most Scenic Route Ever

Early in the game you need to rent a moped. The first thing you do is to go to a museum and pick up a cap. Next, you go behind a church and pet the cat there. It will run through a hole in a nearby shed door. When it comes out, it sits on a ledge. You wait while the guy in the nearby cemetery who is misting his plants leaves, then you take his sprayer. You put some masking tape on the hole in the door and spray the cat, which makes it run into the shed brushing against the tape which will now have some black fur on it. Your next move is to go to a hotel where your old friend Franklin Mosely is staying. You pick up candy from the lobby and put it underneath a painting, then you return to the lobby and buzz Mosely's room. When Mosely leaves to respond to the buzzer, he'll walk past the candy and want to eat it. You can pickpocket his passport from him while he's getting the candy, so long as he hasn't seen you approach. Then, you can go to his room and steal his coat.

Here comes the interesting part. You return to the lobby and take a black pen from the desk. You then go to the dining room and take some syrup. You combine the syrup with the fur off the masking tape to make a false moustache. You put on the moustache, the cap from the museum and the coat you stole to disguise yourself as Mosely. As Mosely doesn't have a moustache, you draw one on his passport with the pen. This enables you to rent the moped you need.

You saw that, right? You have to collect fur from a cat and mix it with syrup to make a false moustache so you can disguise yourself as someone who doesn't have a moustache. You have to draw one on his passport.

This is why adventure games were no longer played. This is what fate awaits casual games if they also take the same-but-harder approach to sophistication. You're not going to do that, right?

Richard Bartle



Figure 7 »Gabriel Knight 3« was one of the last and most critically acclaimed adventure games of the late 90ies.

Win WinWin Win



13.-17.08.2014
Köln

Jetzt Tickets sichern!
gamescom.de

Koelnmesse GmbH

Messeplatz 1, 50679 Köln, Deutschland

Telefon: +49 180 60 89 999*, gamescom@visitor.koelnmesse.de

(*0,20 Euro/Anruf aus dem dt. Festnetz; max.0,60 Euro/Anruf aus dem Mobilfunknetz)



gamescom

Celebrate the games!

BIU
Bundesverband Interaktive
Unterhaltungssoftware e.V.

 **koelnmesse**

WWW.makinggames.de

Online-Register mit Suchfunktion. Zu jedem Key Player finden Sie zusätzlich ein detailliertes Firmenportrait.

Helfen Sie uns!

Sie haben einen Fehler entdeckt oder vermissen Ihre Firma? Dann schreiben Sie uns:

info@makinggames.de, Betreff: Firmenregister

DIE BRANCHE IM ÜBERBLICK FIRMENREGISTER

Vergessen Sie die Auskunft! Hier finden Sie sämtliche uns bekannten Publisher, Entwickler, Dienstleister, Ausbilder und Verbände der deutschsprachigen Spiele-Industrie - von 3d-io bis Z-Software.

DEVELOPMENT					
Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail	
3d-io	Rheingaustraße 94	65203 Wiesbaden	0611 / 5829602	info@3d-io.com	
5d lab	Georges-Köhler-Allee 79	79110 Freiburg	0761 / 2038043	info@5dlab.com	
Adrastea	Bielefelder Str. 5	90425 Nürnberg	0700 / 42300000	info@adrastea.com	
ANIMA Entertainment	Konsul-Smidt-Straße 84	28217 Bremen	0421 / 5784220	info@anima-entertainment.de	
APP33	Fuggerstraße 12	84419 Schwindegg	032 / 221476589	info@app33.de	
Avaloop IT Solutions	Hütteldorferstraße 253	A-1140 Wien		office@avaloop.com	
B-Alive	Biberacher Straße 38	88444 Ummendorf	07351 / 441307	admin@b-alive.de	
BigBlackBlock Gamestudio	Hohenbudberger Straße 133	47229 Duisburg	02065 / 9609747	info@big-black-block.com	
BIGITEC	Theaterstraße 22	53111 Bonn	0228 / 42976585	info@bigitec.com	
BIGPOINT	Drehbahn 47-48	20354 Hamburg	040 / 8814130	info@bigpoint.net	
binary madness	Bundesallee 55	10715 Berlin	030 / 25093746	info@binarymadness.de	
Bit Barons c/o Wailua Games	Schellingstraße 35	80799 München	089 / 41617744	info@bitbarons.com	
Bitfield	Rigaer Straße 56	10247 Berlin	030 / 40045181	kontakt@bitfield.de	
Black Forest Games	Innovationszentrum, Hauptstraße 1a	77652 Offenburg	0781 / 31057910	info@bfgames.biz	
Blue Byte	Adlerstraße 74	40211 Düsseldorf	0211 / 338000	info@bluebyte.de	
Blue Byte Mainz	Römerpassage 1	55116 Mainz	06131 / 554470	info@related-designs.de	
BlurredVision	Glashütter Weg 80	22844 Norderstedt	040 / 53532683	info@blurredvision.de	
Bongfish	Schönaugasse 8a/2	A-8010 Graz	+43 / 676 / 7822203	info@bongfish.com	
Brightside Games	Marie-Elisabeth-Lüders-Straße 1	10625 Berlin	0177 / 3354447		
Broken Rules Interactive Media	Johann Strauss Gasse 20/8	A-1040 Wien		hello@brokenrul.es	
Bronx Studios Entertainment	Olof-Palme-Straße 13	60439 Frankfurt	069 / 870050660	info@bronxstudios.com	
BSS web consulting	Savignystraße 74	45147 Essen	0201 / 5200044	info@bss-consulting.de	
Buntware Games	Colmarerstrasse 9	79106 Freiburg	0761 / 3891598	contact@buntware.com	
BURNS Games	Danziger Straße 118	10405 Berlin	030 / 40301763	info@burns-games.com	
Bright Future	Im Zollhafen 15-17	50678 Köln	0221 / 78982171	info@brightfuture.de	
Bumblebee.	Gerresheimer Straße 7	51674 Wiehl	02261 / 4795001	tassilo@bumblebee-games.com	
ByteRockers	Schönhauser Allee 36	10435 Berlin	030 / 44318610	info@byterockers.de	
Bytro Labs	Rödingsmarkt 29	20459 Hamburg		info@bytro.com	
Caipirinha Games	Kiefernweg 1a	24161 Altenholz	0431 / 324540	info@caipirinhagames.de	
Candygun Games	Hamburger Strasse 132	22083 Hamburg	040 / 20947878		
cerasus.media	Rosenthaler Straße 51	10178 Berlin	030 / 802089990	info@cerasus.de	
Chasing Carrots	Hornbergstraße 94	70188 Stuttgart	0711 / 64519353	office@chasing-carrots.com	
Chimera Entertainment	Sendlinger-Tor-Platz 6	80336 München	089 / 210205790	info@chimera-entertainment.com	
CipSoft	Gabelsbergerstraße 11	93047 Regensburg	0941 / 6308280	kontakt@cipsoft.com	
Claw Design	Ludwigstraße 8	72474 Winterlingen	07434 / 3931	info@clawdesign.com	
Cliffhanger Productions	Seidengasse 32/NB/7	A-1070 Wien		office@cliffhangerproductions.com	
ClockStone STUDIO	Claudiastraße 14	6020 Innsbruck	+43 / 512 / 890377	info@clockstone.com	
Cooee	Europaallee 7-9	67657 Kaiserslautern	0900 / 1002273		
Crafty Studios	Lichtenwörthergasse 112a	A-7033 Pötsching		info@craftystudios.com	
createale	Mörkestraße 28/2	71636 Ludwigsburg	07141 / 2399563	contact@createale.de	
Createm Software	Sunderup 6	24943 Flensburg	0461 / 64808		
Crenetic	Kreuzstraße 3	45468 Mülheim an der Ruhr	0208 / 4442142	info@crenetic.de	
Crytek	Grüneburgweg 16-18	60322 Frankfurt am Main	069 / 21977660	info@crytek.de	
Cyber Arena Entertainment Group	Sebastian Kneippgasse 30	2380 Perchtoldsdorf		info@c-arena.com	
Dachshund Games	Turmstraße 48	10551 Berlin		team@dachshund-games.com	
Daedalic Entertainment	Papenreye 51	22453 Hamburg	040 / 43261270	info@daedalic.de	
Decane	Hallerstraße 5e	20146 Hamburg	040 / 42910947	info@decane.net	
Deck13 Interactive	Gutleutstraße 82	60329 Frankfurt am Main	069 / 71671660	info@deck13.com	
Deep Silver Fishlabs	Großer Burstah 50-52	20457 Hamburg	040 / 8888000	business@fishlabs.net	
Destrax Entertainment	Merkurstraße 10	33739 Bielefeld		info@destrax.de	
Digitalmindsoft	Wichernstraße 10	89073 Ulm		info@digitalmindsoft.eu	
Doublesmith	Ranzonigasse 3/3	A-3100 St. Pölten	+43 / 699 / 19447880	mail@doublesmith.com	
dreINET	Bergdorfer Straße 96	12623 Berlin		philip@dreinet.org	
DynamicHead	Hauptstraße 6	55283 Nierstein Schwabsburg	061 / 33572494	timo.weisse@dynamichead.com	
eelusion	Paul-Lincke-Ufer 39	10999 Berlin		contact@eelusion.com	
Egosoft	Heidestraße 4	52146 Würselen	02405 / 4239970	egoinfo@egosoft.com	
Enter-Brain-Ment	Littenstraße 106/107	10179 Berlin		interesse@enter-brain-ment.com	
Envision Entertainment	Binger Straße 38	55218 Ingelheim		info@envision-entertainment.de	
eos interactive	Carl-Zeller Straße 16	75015 Berlin	07252 / 957757	info@eosinteractive.de	
exDream	Hildesheimer Straße 5	30169 Hannover	0511 / 37077675	info@exdream.com	
exozet	Platz der Luftbrücke 4-6	12101 Berlin	030 / 2465600	games@exozet.com	
extra toxic	Hanauer Landstraße 196a	60314 Frankfurt am Main	069 / 48440040	info@extratoxic.com	
eyefactive	Feldstraße 128	22880 Wedel	04103 / 903800	info@eyefactive.com	
FAKT Software	Dufourstraße 28	04107 Leipzig	0341 / 14990564	info@fakt-software.de	

DEVELOPMENT					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Fameside Entertainment		Storkower Straße 158	10407 Berlin	030 / 40577053	info@fameside.com
Fancy Bytes		Niederstraße 5	26931 Huntorf	02151 / 3259912	biz@fancy-bytes.de
FDG Entertainment		Agnes-Pockels-Bogen 1	80992 München	089 / 18941370	info@fdg-entertainment.com
flaregames		Kaiserstraße 146-148	76133 Karlsruhe	0721 / 1831540	info@flaregames.com
Freudenreich & Tollknaepper		Paracelsusstraße 57	42283 Wuppertal	0202 / 2721048	mail@german-railroads.com
Funatics Software	key player	Essener Straße 66	46047 Oberhausen	0208 / 3066670	info@funatics.de
funworld		Photo Play Straße 1	A-4860 Lenzing	+43 / 7662 / 67850	info@photoplay.com
Fusionsphere Systems		Rosenau 8	86551 Aichach	0700 / 38746600	info@fusionsphere.com
GameArt Studio		Bismarckstraße 107	10625 Berlin		kontakt@GameArtStudio.de
GameDuell	key player	Taubenstraße 24-25	10117 Berlin	030 / 288768210	impressum@gameduell.de
Gameforge		Albert-Nestler-Straße 8	76131 Karlsruhe	0721 / 3548080	info@gameforge.de
gameus		Kufsteinerstraße 40	83064 Raubling	08035 / 6689	info@gameus.de
Gaming Minds Studios		Georgstraße 6	33330 Gütersloh		contact@gamingmindsstudios.com
Gbanga		Brauerstraße 31	8004 Zürich	+41 / 43 / 5366701	matthias@gbanga.com
GIANTS Software GmbH		Rütistrasse 18	8952 Zürich-Schlieren	+41 / 44 / 5082176	mail_us@giants-software.com
Goal Games		Feurigstraße 54	10827 Berlin		info@goal-games.de
goodbeans		Winsstraße 62	10405 Berlin	030 / 802087400	jobs@goodbeans.com
Goodgame Studios	key player	Theodorstraße 42-90, Haus 9	22761 Hamburg	040 / 2198800	pr@goodgamestudios.com
Greentube		Mariahilfer Straße 47/1/102	A-1060 Wien	+43 / 1 / 4945056	office@greentube.com
HandyGames		i Park Klingholz 13	97232 Giebelstadt	09334 / 97570	info@handy-games.com
iLogos Europe	key player	Pezoldamm 98a	22175 Hamburg	0160 / 7002927	alexander@ilogos.biz
Independent Arts Software	key player	Münsterstraße 5 / HAMTEC Haus 2b	59065 Hamm	02381 / 488818	info@independent-arts-software.de
InnoGames	key player	Friesenstraße 13	20097 Hamburg	040 / 78893350	info@innogames.de
Instance Four		Friederikastraße 65	44789 Bochum	0234 / 89038971	info@instancefour.com
Island Games		Vahrenwalder Straße 7	30165 Hannover	0511 / 35319898	kontakt@islandgames.de
Jo-Mei		Monumentenstraße 35, Aufg. C	10829 Berlin	030 / 81494080	info@jo-mei.com
K1010 Media		Erkelenzdamm 59-61	10999 Berlin	030 / 57709336	info@k1010.de
keen games		Dreieichstraße 59	60594 Frankfurt am Main	069 / 66058838	info@keengames.com
King Art		Martinstraße 47-49	28195 Bremen	0421 / 322760	contact@kingart.de
KORION Simulation		Mömpelgardstraße 16	71640 Ludwigsburg	07141 / 3898231	info@korion.de
kr3m.media		Kaiserstraße 158	76133 Karlsruhe	0721 / 18395918	
Kritzelkrat3 3000		Dieselstraße 2-6	97082 Würzburg	0931 / 3536091	info@kritzelkrat3.de
kunst-stoff		Kottbusser Damm 73	10967 Berlin	030 / 23457178	info@kunst-stoff.de
Limbic Entertainment		Heinrich-Hertz-Straße 21	63225 Langen	06103 / 5019780	info@limbic-entertainment.de
MADE		Alter Schlachthof 39	76131 Karlsruhe	0721 / 78809460	
MaDe Games GbR Neetix - Wycislik		Hochstraße 34	41334 Nettetal		info@madegames.com
marune		Wallbergstraße 16	82008 Unterhaching	089 / 66500749	info@marune.de
McPeppergames		Nikolaus-Molitor-Straße 11a	97702 Münnerstadt	09733 / 781006	info@mcpeppergames.com
Media Seasons		Liviastraße 6	4105 Leipzig	0341 / 2156752	contact@mediasaisons.com
Mimimi Productions		Berg-am-Laim-Straße 64	81673 München	089 / 809115070	info@mimimi-productions.de
Mind Elevator Games		Prenzlauer Promenade 190	13189 Berlin	030 / 91744394	info@mindelevatorgames.com
mobivention		Gottfried-Hagen-Straße 24	51105 Köln	0221 / 6778110	info@mobivention.com
Moonbyte Games		Wormser Landstraße 17	67346 Speyer	06232 / 8774482	info@moonbytegames.com
NeoBird		Südwestpark 37-41	90449 Nürnberg	0911 / 3092097	info@neobird.de
netmin games		Philipp-Reis-Straße 6	55129 Mainz	06131 / 507896	schreiber@netmin.de
Neutron Games		Gustav-Meyer-Allee 25	13355 Berlin	030 / 46307222	kontakt@neutron-games.de
northworks		Stahlwiete 23	22761 Hamburg	040 / 78107075	info@northworks.de
Noumena Studios		Lützowstraße 33	10785 Berlin	030 / 53148825	info@noumenastudios.com
NovaTriX / Animedo		Schlüsselwiesen 19	70186 Stuttgart	0711 / 3102770	info@animedo.de
Nurogames GmbH		Schaaftenstraße 25	50676 Köln	0221 / 3988040	info@nurogames.com
OG-Soft Productions		Zirbelstraße 48	86154 Augsburg		info@og-soft.de
online4ever Limited		Manitiusstraße 6	1067 Dresden	0351 / 2063875	info@online4ever.biz
OFM - OnlineFussballManager	key player	Eupener Straße 60	50933 Köln	0221 / 88892920	office@onlinefussballmanager.de
Outline Development		Hauptstraße 122	57074 Siegen	0271 / 3131331	webmaster@outline-development.de
Pappuga		Fürther Straße 212	90429 Nürnberg	0911 / 32378310	info@pappuga.com
Piece of Pie Studios		Sendlinger-Tor-Platz 6	80336 München	089 / 210205780	info@pieceofpiestudios.com
Piranha Bytes		Ruhrallee 63	45138 Essen	0201 / 806720	info.nospam@piranha-bytes.com
PixelPlaying		Vorm Eickerberg 1a	42929 Wermelskirchen	02196 / 8899693	info@pixelplaying.de
pixeltamer.net		Clausewitzstraße 6	10629 Berlin	030 / 34347690	welcome@pixeltamer.net
Pixon		Jaffestraße 10	21109 Hamburg	040 / 33399430	welcome@pixon.de
Playnik		Bielefelder Str. 5	90425 Nürnberg	0700 / 42300000	info@adrastea.com
playzo		Heinrich-Herz- Straße 6	64295 Darmstadt	06151 / 6673700	info@playzo.biz
plazz entertainment		Augustmauer 1	99084 Erfurt	0361 / 2169460	info@plazz-entertainment.com
Pop Rocket Games		Gasstraße 10	22761 Hamburg	040 / 822449040	
Procontis		Am Bauhof 18	64807 Dieburg	06071 / 430111	sales@procontis.de
Promotion Software		Karlstraße 3	72072 Tübingen	07071 / 91670	mail@promotion-software.de
qforge		Südwestpark 37 - 41	90449 Nürnberg	0911 / 3092097	office@qforge.de
Quadriga Games		August-Bebel-Straße 27	14482 Potsdam	0331 / 23189660	info@quadrigagames.com
Rabenaue		Ebelingstraße 17	10249 Berlin	030 / 21092345	the_contact@rabenaue.com
Rainfire Entertainment		Frohnhauserstraße 65	45127 Essen	0201 / 94613403	info@rainfire-entertainment.com
Rat King Entertainment		Robert-Franz-Ring 3	06108 Halle	0345 / 614741	mail@ratking.de
Ravensburger Digital		Kaflerstraße 8	81241 München	089 / 12228180	tellme@ravensburger-digital.com
Reality Twist		Berg-Am-Laim-Straße 64	81673 München	089 / 380129500	mail@reality-twist.com
Realmforge Studios		Hofer Straße 15	81737 München	089 / 55069134	info@realmforgestudios.com
RedMoon Studios		Dieselstrasse 16	86368 Gersthofen	0821 / 45598000	info@redmoonstudios.de
remote control productions		Sendlinger-Tor-Platz 6	80336 München	089 / 21020570	info@r-control.de
RockABYTE	key player	Schaaftenstraße 25	50676 Köln	0221 / 80147950	info@rockabyte.com
SAFKAS Development		Schulze-Delitzsch-Straße 39	04315 Leipzig		mail@safkas.de
Scorpius Forge	key player	Nelly-Sachs-Straße 119	89134 Blaustein	0731 / 14410099	info@scorpius-forge.de
Serious Games Solutions		August-Bebel-Straße 27	14482 Potsdam	0331 / 23189680	mail@serious-games-solutions.com
SharkbomB Studios		Durlacher Allee 53	76131 Karlsruhe	0160 / 92502633	contact@sharkbombs.de
SilentFuture		Triebelsheide 37b	42111 Wuppertal	0202 / 3935366	info@silentfuture.de
Silver Seed Games		Sandtorstraße 23	39106 Magdeburg	0391 / 5448619372	info@silverseedgames.com
Simlity		Moritz-von-Rohr-Straße 1a	07745 Jena	03641/3279684	info@simlity.com
Skunk Brothers		Dieselstraße 12	61191 Rosbach		hello@skunkbrothers.de

DEVELOPMENT					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
SlipShift		Friedrichstraße 122	10117 Berlin	030 / 27581581	info@slipshift.net
Snowstep Development		Heinrich-Holtzschneider-Weg 76	40489 Düsseldorf	0201 / 17848176	business@snowstep.com
Socialspiel		Schmalzhofgasse 26 / Top 5-6	A-1060 Wien		office@socialspiel.com
Solimedia Productions		Bahnhofstr. 4a	99084 Erfurt	0361 / 658550	hatmut.niemer@solimedia.de
Sparrow Games		Wegenerstraße 1	13088 Berlin	030 / 50915356	mail@sparrowgames.de
SpinBottle Games		Weidkamp 3	45355 Essen	0201 / 1789614	info@spinbottlegames.com
Spirit		Friedrichstraße 15	70174 Stuttgart	0711 / 96685510	weyrich@go4spirit.com
Splitscreen Studios		Neuer Kamp 30	20357 Hamburg		info@splitscreenstudios.com
Sportradar Virtual Gaming		Konsul-Smidt-Straße 8f	28217 Bremen	0421 / 6195640	info@aitainment.de
Springwald Software		Alter Eistreff 36	44789 Bochum	0700 / 777464925	info@springwald.de
Sproing Interactive Media	key player	Fernkornegasse 10	A-1100 Wien	+43 / 1 / 60430280	office@sproing.com
Studio Fizbin		Hoferstraße 20	71636 Ludwigsburg	07141 / 9923866	hello@studio-fizbin.de
Sunlight Games		Im Klapperhof 7-23	50670 Köln	0221 / 16823466	kontakt@sunlight-games.com
symlCrowd		In der Schleif 27	5224 Eschweiler	0152 / 55878044	apps@symlcrowd.de
Synetic		Auf der Haar 47	33332 Gütersloh	05241 / 4009900	synetic@synetic.de
Team Vienna Games		Kaiserstrasse 84/1/6	A-1070 Wien		office@teamviennagames.com
Travian Games		Wilhelm-Wagenfeld-Straße 22	80807 München	089 / 3249150	mail@traviangames.com
unikat media		Eckdrift 10	19061 Schwerin	0385 / 64105688	info@unikatmedia.de
upjers	key player	Hafenstraße 13	96052 Bamberg	0951 / 510908100	mail@upjers.com
VisionaryX		Friedrichstraße 12	71101 Schönaich	07031 / 2850 / 414	contact@visionaryx.com
V-Play		Kolonitzgasse 9/11-14	A-1030 Wien		team@v-play.net
Vulcando Games		Dasinger Straße 2	86165 Augsburg	0821 / 4550834	info@vulcando-games.de
Wolpertinger Games		Sendlinger-Tor-Platz 6	80336 München	089 / 210205700	contact@wolpertingergames.com
wooga		Saarbrücker Straße 38	10405 Berlin	030 / 32505294	info@wooga.com
Xendex		Mariahilfer Strasse 176/6	A-1150 Wien	+43 / 1 / 89080090	office@xendex.com
XYRALITY		Friedensallee 290	22763 Hamburg	0 / 4035730010	info@xyrality.com
YAGER Development	key player	Pfuehlstraße 5	10997 Berlin	030 / 69597660	contact@yager.de
zeitland media & games		Königsallee 43	71638 Ludwigsburg	07141 / 5050565	contact@zeitland.com
Zeroscale		Schwedter Straße 9a	10119 Berlin	030 / 405057929	info@zeroscale.de
Zone 2 Connect		Fürstenwall 70	40219 Düsseldorf	0211 / 5422400	info@zone2.de
Z-Software		Lindemannstrasse 81	44137 Dortmund	0231 / 33015031	contact@z-software.net
FREELANCER DEVELOPMENT					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Caspar Siebel		Theresienhöhe 1	80339 München	017661615291	mail@casparsiebel.com
Eric Jannot (Game Design, Game Writing)		Sonnenallee 70	12045 Berlin	0178 / 5223697	
Jochen Hamma (Producing)		Wehinger Weg 9	78583 Boettingen	07429 / 910793	jochen.hamma@fantastic-realms.com
Jörg Beilschmidt (Game Design)		Berliner Ring 20	21406 Melbeck	0172 / 4311243	j.beilschmidt@creatown.de
Kim Lange (Full Service Agency)		Landwehr 192	46045 Oberhausen	0178 / 2375418	lange@eins2design.de
Marco Sowa (Producing, Development)		Postfach 420566	50899 Köln	0221 / 16916844	mail@marcosowa.de
Mariusz Ginel (Mediengestaltung, SED)		Stettiner Straße 31	53840 Troisdorf	02241/9321556	mginel@mg-mediengestaltung.de
Martin Linnartz (3D Artist, Game Developer)		Bleichstraße 12	41460 Neuss	02131 / 3133306	
Martin Nerurkar (Game Design)		Durlacher Allee 53	76131 Karlsruhe		mnerurkar@sharkbombs.com
Nico Nowarra (Storytelling)		Friedbergstraße 9	14057 Berlin	030 / 31801899	nno@menintext.com
Niels Bauer Games (Game Design)		Sickingenstraße 37	79117 Freiburg	0761 / 66947	contact@nielsbauergames.com
Thomas Obermaier (Programmierung)		Kapellenstraße 1	86869 Gutenberg	0170 / 3815144	contact@polyforming.com
Wolfgang Walk (Full Scale Producing)		Karl-Leopold-Straße 6	76229 Karlsruhe	0721 / 48399963	wolfgang.walk@t-online.de
PUBLISHING/DISTRIBUTION					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
505 Games		Brunnfeld 2-6	93133 Burglengenfeld	09471 / 3088792	
Activision Blizzard Deutschland		Fraunhoferstraße 7	85737 Ismaning	089 / 9998170	info@activision.de
ad2games	key player	Rosenstraße 17	10178 Berlin	030 / 240888221	info@ad2games.com
Aeria Games Europe		Schlesische Straße 27, Aufgang C	10997 Berlin		info@aeriagames.com
Aerosoft		Lindberghring 12	33142 Bielefeld	02955 / 760310	info@aerosoft.de
Application Systems Heidelberg		Pleikartsforsterhof 4/1	69124 Heidelberg	06221 / 300002	kontakt@application-systems.de
AppLift	key player	Rosentraße 17	10178 Berlin	030 / 240888208	info@applift.com
BHV		Novesiasstraße 60	41564 Kaarst	02131 / 76501	info@bhv.de
Bigpoint	key player	Drehbahn 47-48	20354 Hamburg	040 / 881430	info@bigpoint.net
bitComposer Entertainment	key player	Mergenthalerallee 79-81	65760 Eschborn	06196 / 7793810	info@bit-composer.com
Capcom - CEG Interactive Entertainment		Barmbeker Straße 4b	22303 Hamburg	040 / 6965620	info@capcomeuro.de
Covus Crobo	key player	Schwedter Straße 263	10119 Berlin		info@crobo.com
Crimson Cow		Tarpen 40, Haus 5	22419 Hamburg	040 / 59355237	info@crimsoncow.de
Daedalic Entertainment	key player	Papenreye 53-Workport Unit 2	22453 Hamburg	040 / 43261270	info@daedalic.de
Deutsche Telekom	key player	T-Online-Allee 1	64295 Darmstadt	06151 / 6800	presse@gamesload.de
Disney Interactive Studios		Kronstadter Straße 9	81677 München	089 / 993400	info@disney.com
dreamfab		Domplatz 3	93047 Regensburg	0941 / 569591520	
dtp entertainment		Goldbekplatz 3-5	22303 Hamburg	040 / 6699100	info@dtp-entertainment.com
dtp young entertainment		Goldbekplatz 3-5	22303 Hamburg	040 / 6699100	info@dtp-young.com
Electronic Arts		Im Zollhafen 15-17	50678 Köln	0221 / 975820	
European Games Group		Wiener Platz 7	81667 München		contact@gamesgroup.eu
EuroVideo Bildprogramm		Oskar-Messter-Straße 15	85737 Ismaning	089 / 9624440	info@eurovideo.de
flashpoint	key player	Valvo Park - Haus 5c / Tarpen 40	22419 Hamburg	040 / 710060	info@flashpoint.de
GameDuell		Taubenstraße 24-25	10117 Berlin	030 / 288768211	impressum@gameduell.de
Gameforge	key player	Albert-Nestler-Straße 8	76131 Karlsruhe	0721 / 3548080	info@gameforge.de
GameGenetics	key player	Alte Jakobstraße 85-86	10179 Berlin	030 / 922512769	info@gamegenetics.de
GameLoft		Karl-Liebknecht-Straße 5	10118 Berlin	030 / 20188429	gregory.wintgens@gameloft.com
GameTwist		Mariahilfer Straße 47/1/102	10600 Wien	+43 / 1494 / 5056	3rdparty@gametwist.com
gamigo	key player	Behringstraße 16b	22765 Hamburg	040 / 4118850	info@gamigo.com
GFACE	key player	Grüneburgweg 16-18	60322 Frankfurt am Main	069 / 21977660	cryengine@crytek.com
Halycon Media		An der Autobahn 24	23858 Reinfeld	04533 / 61090	info@halycon.de
Headup Games	key player	Nordstraße 102	52353 Düren	02421 / 4868700	info@headupgames.com
HitFox Group	key player	Rosenstraße 17	10178 Berlin	030 / 240888200	info@hitfoxgroup.com
Infernum Productions	key player	Uhlandstraße 175	10719 Berlin	030 / 91201090	contact@infernum.com
InnoGames	key player	Friesenstraße 13	20097 Hamburg	040 / 78893350	info@innogames.de

PUBLISHING/DISTRIBUTION					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Intenium		Neuer Pferdemarkt 1	20359 Hamburg	040 / 25494296	info@intenium.de
Kalypso Media Group	hex player	Wilhelm-Leuschner-Straße 11-13	67547 Worms	06241 / 5061900	info@kalypsomedia.com
Koch Media	hex player	Lochamer Straße 9	82152 Planegg/München	089 / 242450	office@kochmedia.de
Konami		Berner Straße 103-105	60437 Frankfurt am Main	069 / 9855730	support@konami.de
McGame.com		Fürstenwall 69	40217 Düsseldorf	0211 / 63550100	support@mcgame.com
Microsoft		Konrad-Zuse-Straße 1	85716 Unterschleißheim	089 / 31760	kunden@microsoft.com
Millform		Ringstraße 110	CH-4106 Therwil	+41 / 61 / 7220712	werner.sala@millform.ch
Morphicon Limited	hex player	Lindwurmstraße 88	80337 München	089 / 45235460	info@morphicon.de
MMM MultiMediaManufaktur		Ludwigstraße 33-37	60327 Frankfurt am Main	069 / 974600	info@multimediamanufaktur.com
Namco Bandai Partners		Carl-Benz-Straße 21	60386 Frankfurt am Main	069 / 24449100	de.info@namcobandaipartners.com
NBG		Brunnfeld 2-6	93133 Burglengenfeld	09471 / 70170	info@nbg-online.de
Nintendo		Nintendo Center	63760 Grossostheim	06026 / 9500	info@nintendo.de
Novitas		Frankfurter Straße 2	38122 Braunschweig	02574 / 9270	kontakt@greenpepper.de
Phenomenia		Friedrich-Ebert-Straße 14	44866 Bochum	02327 / 9970	info@phenomenia.com
Playa Games	hex player	Alstertor 9	20095 Hamburg	040 / 76793220	info@playa-games.com
Players Rock Entertainment		Goseriede 4	30159 Hannover	0511 / 3539940	contact@playersrock.com
ProSiebenSat.1 Games	hex player	Medienallee 6	85774 Unterföhring	089 / 950710	info@sevengames.de
qforge	hex player	Südwestpark 37 - 41	90449 Nürnberg	0911 / 3092097	info@qforge.de
Ravensburger Digital	hex player	Kaflerstraße 8	81241 München	089 / 12228180	tellme@ravensburger-digital.com
Reality Twist		Berg-Am-Laim-Straße 64	81673 München	089 / 380129500	mail@reality-twist.com
redspotgames		Postfach 310464	80104 München	089 / 4036370	info@redspotgames.com
Rondomedia		Limitenstraße 64-78	41236 Mönchengladbach	02166 / 618660	info@rondomedia.de
RTL Games		Achnerstraße 1036	50858 Köln	0221 / 4560	entwickler@rtl-interactive.de
Rumble Media		Kriegsstraße 5	76137 Karlsruhe	0721 / 2018020	info@fettspielen.de
S.A.D.		Rötelbachstraße 91	89079 Ulm	07305 / 96290	kontakt@s-a-d.de
Sega		Rablstraße 24	81669 München	089 / 22848000	info@sega.de
Sony Computer Entertainment		Frankfurter Straße 233	Neu-Isenburg	06102 / 4330	scee_presscentre@scee.net
Square Enix		Domstraße 17	20095 Hamburg	040 / 30633400	
Take 2		Agnesstraße 14	80798 München	089 / 278220	info@take2.de
Thumbr		Pfuehlstraße 5	10997 Berlin	030 / 300137100	info@thumbr.com
Tipp24 Entertainment		Straßenbahnhof 11	20251 Hamburg		info@t24e.de
Tivola Publishing		Oeverseestraße 10-12	22769 Hamburg	030 / 5363580	mail@tivola.de
Traffic Captain		Wendenstraße 21b	20097 Hamburg	040 / 23706800	info@trafficcaptain.com
Travian Games	hex player	Wilhelm-Wagenfeld-Straße 22	80807 München	089 / 3249150	mail@traviangames.com
Ubisoft	hex player	Adlerstraße 74	40211 Düsseldorf	0211 / 338000	info@ubisoft.de
United Soft Media Verlag		Thomas-Wimmer-Ring 11	80539 München	089 / 29088175	info@usm.de
upjers	hex player	Hafenstraße 13	96052 Bamberg	0951 / 510908100	mail@upjers.com
ZeniMax Germany		Schillerstraße 15/17	60313 Frankfurt am Main		kontakt.de@zenimax.com
Zuxxez		Rittnerstraße 36	76227 Karlsruhe	0721 / 464720	info@zuxxez.com
PUBLIC SECTOR					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
BIU	hex player	Rungestraße 18	10179 Berlin	030 / 24087790	info@biu-online.de
FilmFernsehsFonds Bayern	hex player	Sonnenstraße 21	80331 München	089 / 5446020	filmfoerderung@fff-bayern.de
G.A.M.E.	hex player	Windscheidstraße 18	10627 Berlin	030 / 346550980	info@game-bundesverband.de
gamearea FRM		Holzhausenstraße 22	60322 Frankfurt am Main	069 / 94419441	info@gamearea-frm.de
gamecity:Hamburg	hex player	Habichtstraße 41	22305 Hamburg	040 / 2270190	info@gamecity-hamburg.de
GAMEplaces		Hanauer Landstraße 126-128	60314 Frankfurt am Main	069 / 21236214	manuela.schiffner@gameplaces.de
games.net berlinbrandenburg	hex player	Wöhlerstraße 12 - 13	10115 Berlin	030 / 24628570	games.net@medianet-bb.de
GDI.Ruhr		Kreuzstraße 1	45468 Mülheim an der Ruhr	0208 / 69801520	info@games-ruhr.com
Hessen-IT		Abraham-Lincoln-Straße 38 - 42	65189 Wiesbaden	0611 / 7748481	info@hessen-it.de
medienboard berlin-brandenburg	hex player	August-Bebel-Straße 26-53	14482 Potsdam-Babelsberg	0331 / 743870	info@medienboard.de
Mitteldeutsche Medienförderung		Hainstraße 17-19	4109 Leipzig	0341 / 269870	info@mdm-online.de
nordmedia	hex player	Expo-Plaza 1	30539 Hannover	0511 / 1234560	info@nordmedia.de
Northstar Developers		Bödekerstr. 84	30161 Hannover	0160 / 97880272	d.brueinig@collision-studios.com
ÖVUS		Aichhornstraße 14	A-1120 Wien	+43 / 1810 / 328912	office@ovus.at
Stiftung Digitale Spielkultur	hex player	Torstraße 6	10119 Berlin	030 / 29049290	kontakt@stiftung-digitale-spielkultur.de
USK	hex player	Torstraße 6	10119 Berlin	030 / 24088660	kontakt@usk.de
WERK1 München	hex player	Grafinger Straße 6	81671 München	089 / 9952990	info@werk1muenchen.de
EDUCATION					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Acagamics e.V.		Universitätsplatz 2, G29	39106 Magdeburg		info@acagamics.de
Akademie für Medien		Balinger Straße 39A	70567 Stuttgart	0711 / 4209444	h.ehlers@amk-net.de
Cologne Game Lab / Fachhochschule Köln		Ubierring 40	50678 Köln	0221 / 82753095	info@colognegamelab.de
Designhochschule Schwerin		Bergstraße 38	19055 Schwerin	0385 / 5559775	info@designschule.de
EC Europa Campus		Lyonerstraße 34	60528 Frankfurt am Main	0621 / 8425660	
FH OÖ Studienbetriebs GmbH		Softwarepark 11	A-4232 Hagenberg	+43 (0) 5080420	info@fh-hagenberg.at
FH Trier		Schneidershof	54293 Trier	0651 / 8103345	sek@informatik.fh-trier.de
FH Technikum Wien		Mariahilfer Straße 37-39	1060 Wien	+43 / 1 / 588390	info@technikum-wien.at
FH Salzburg		Urstein Süd 1	A-5412 Puch/Hallein	+43 50 / 22110	press@fh-salzburg.ac.at
Games Academy Berlin	hex player	Rungestrasse 20	10179 Berlin	030 / 29779120	info@games-academy.de
Games Academy Frankfurt	hex player	Hanauer Landstraße 146	60314 Frankfurt	069 / 42696460	info@games-academy.de
Gesellschaft für Personalentwicklung und Bildung		Beuthstraße 7-8	10117 Berlin	030 / 9339480	gpb-berlin@gpb-berlin.deHAW Hamburg
Hochschule Mittweida (FH)		Technikumplatz 17	09648 Mittweida		kontakt@hs-mittweida.de
HTW Berlin		Wilhelminenhofstraße 75a	12459 Berlin	030 / 50190	
Macromedia Akademie		Eisenheimerstraße 63	80687 München	089 / 96160800	ausbildung@macromedia.de
Macromedia Akademie		Naststraße 11	70376 Stuttgart	0711 / 2807380	info.stgt@macromedia.de
Macromedia Akademie		Brüderstraße 17	50667 Köln	0221 / 3108223	akademie.koeln@macromedia.de
Macromedia Hochschule		Gollnerstraße 4	80339 München	089 / 5441510	info.muc@macromedia.de
Mediadesign Hochschule		Lindenstraße 20-25	10969 Berlin	030 / 3992660	info-ber@mediadesign-fh.de
Mediadesign Hochschule		Werdener Straße 4	40227 Düsseldorf	0211 / 1793330	info-dus@mediadesign-fh.de
Mediadesign Hochschule		Claudius-Keller-Straße 7	81669 München	089 / 4506050	info-muc@mediadesign-fh.de
S4G School for Games		Gubener Straße 47	10243 Berlin	030 / 96595244	info@school4games.net
SAE Institute Berlin	hex player	Soltauer Straße 18-22	13509 Berlin	030 / 43094470	berlin@sae.edu

EDUCATION					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
SAE Institute Bochum	key player	Metzstraße 23	44793 Bochum	0234 / 93451310	bochum@sae.edu
SAE Institute Hamburg	key player	Feldstraße 66	20359 Hamburg	040 / 23688080	hamburg@sae.edu
SAE Institute Köln	key player	Medienzentr. Ost, Carlswerkstr. 11c	51063 Köln	0221 / 9541220	koeln@sae.edu
SAE Institute München	key player	Bayerwaldstraße 43	81737 München	089 / 89068771	muenchen.sae.edu
SRH Fachschulen		Bonhoefferstraße 1	69123 Heidelberg	06221 / 884225	it.heidelberg@fachschulen.srh.de
SRH Hochschule Heidelberg		Ludwig-Guttman-Straße 6	69123 Heidelberg	06221 / 882790	Daniel.Goerlich@Hochschule-Heidelberg.de
Zürcher Universität der Künste	key player	Ausstellungsstraße 60	CH-8005 Zürich	+41 / 43 / 4464646	hs.admin@zhdk.ch
BUSINESS SERVICE					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
4-REAL INTERMEDIA (Lokalisation)		Ludwigstraße 32	63067 Offenbach	069 / 8090880	info@4-real.com
Anakan (Lokalisation)		Pfuhlstraße 5	10997 Berlin	030 / 531420450	ana@anakan.de
Aruba Events		Kreuzstraße 1-3	45468 Mülheim an der Ruhr	0208 / 6982608	infos@aruba-events.de
arvato games task force		Carl-Bertelsmann-Straße 161F	33332 Gütersloh	05241 / 803074	kontakt@gamestaskforce.de
ATLAS Interactive Deutschland		Christoph Probst Weg 3	20251 Hamburg	040 / 413300185	sales@atlasinteractive.de
Brehm & v. Moers		Anna-Louisa-Karsch-Straße 2	10178 Berlin	030 / 2603050	berlin@bvm-law.de
Die Hobrechts (Game Thinking)		Hobrechtstraße 65	12047 Berlin	030 / 62901232	info@diehobrechts.de
Dr. Behrmann & Härterl Rechtsanwälte		Kantstraße 98	10627 Berlin	030 / 319984910	info@behrmannhaertel.de
Buschbaum Media & PR		Emanuel-Leutze Straße 21	40547 Düsseldorf	0211 / 5988140	presse@buschbaum-media.com
CULTURETRANSLATE		Lyoner Straße 34	60528 Frankfurt am Main	069 / 659998000	info@culturetranslate.de
deal united		Landshuter Allee 10	80637 München	089 / 416157014	info@dealunited.com
DELASOCIAL		Christoph-Probst-Weg 31	20251 Hamburg	040 / 87979890	contact@delasocial.com
Digital River	key player	Vogelsanger Straße 78	50823 Köln	0221 / 310 88614	sales@digitalriver.com
Effective Media		Josef-Haumann-Straße 10	44866 Bochum	02327 / 22310	info@effective-media.de
elblabs		Bernstorffstraße 120	22767 Hamburg	040 / 76797218	info@elblabs.com
Frankfurter Buchmesse		Reineckstraße 3	60313 Frankfurt am Main	069 / 21020	info@book-fair.com
Freaks 4U Gaming		An der Spreeschanze 10	13599 Berlin	030 / 41719113	info@freaks4u.de
Friedmann Kommunikation		Schanzenstraße 36	51063 Köln	0221 / 5348710	info@fr-k.de
gamescom	key player	Messeplatz 1	50679 Köln	0221 / 8213894	info@koelnmesse.de
Games Foundation		Neuer Kamp 30	20357 Hamburg		info@gamesfoundation.com
Games Quality		Zum Wasserwerk 15	15537 Erkner	03362 / 885688	info@games-quality.com
GAN Game Ad Net		Stresemannstraße 342	22761 Hamburg	040 / 67586750	info@gan-online.com
Ganz & Stock (Personnel Consulting)		Westhafentower, Westhafenplatz	60327 Frankfurt am Main	069 / 710456380	a.stock@ganz-stock.de
Game Developers Conference Europe	key player	Kaiser-Wilhelm-Straße 30	12247 Berlin	030 / 34622644	carsten.kohlenbeck@ubm.com
GAMINSIDE		Kaiserstraße 2-4	1070 Wien	+43 / 0676 / 5379776	info@gaminside.com
Gärtner PR	key player	Destouchesstraße 68	80796 München	089 / 30766854	info@gaertner-pr.de
GRAEF Rechtsanwälte		Jungfrauenhal 8	20149 Hamburg	040 / 80600090	www.graef.eu
HoneyTracks (Game Analytics)		Schelling Straße 35	80799 München	089 / 28723995	info@honeytracks.com
IDG Entertainment Media	key player	Lyonel-Feininger-Straße 26	80807 München	089 / 360860	dbhulaputra@idg.de
IEM Consulting (Business Development, Finanzierung)		Hanauer Landstraße 192	60314 Frankfurt am Main	069 / 15346479	info@iem-consulting.com
IHDE & Partner (Rechtsberatung)		Schoenhauser Allee 10-11	10119 Berlin	030 / 44318660	berlin@onlinelaw.de
IMC (Full-Service Dienstleister)		Hauptstraße 172	51143 Köln	02203 / 949092	info@imc-online.net
INPROMO		Gasstraße 6a	22761 Hamburg	040 / 23881771	kontakt@inpromo.de
iVentureCapital		Wendenstraße 21 B	20099 Hamburg	040 / 8081250	info@iventurecapital.com
Lightning PR		Gundelandsstraße 13	60435 Frankfurt		koetting@lightning-pr.de
Marchsreiter Communications		Guldeinstraße 41a	80339 München	089 / 51919942	dm@marchsreiter.com
Medienachse (Full-Service, Businessplanung)		Trappentreustraße 10	80339 München		info@medienachse.de
M.I.T. - Media Info Transfer		Albert-Einstein-Ring 8	22761 Hamburg	040 / 30066890	info@mediainfotransfer.de
MOTION AREA	key player	Sommerstraße 3	65197 Wiesbaden	0611 / 56597577	info@motionarea.de
Osborne Clarke (Anwaltskanzlei)		Innere Kanalstraße 15	50823 Köln	0221 / 51084000	info@osborneclarke.de
Partnertrans (Lokalisation)		Leipziger Straße 28	40822 Mettmann	02104 / 172660	info@partnertrans.com
paysafecard.com Wertmarken		Am Europlatz 2	1120 Wien	+43 / 1 / 72083800	sales@paysafecard.com
PiXABLE STUDIOS	key player	Buchenstraße 16b	01097 Dresden	0351 / 56341360	info@pixable.de
Pixelworkshop / IG Computergrafik		Hirschengasse 1/26	1060 Wien	+43 / 1 / 588018698	info@pixelworkshop.at
Play + Smile Marketing (PR, Marketing)		Am Wasserturm 6	40668 Meerbusch	02150 / 705660	info@play-and-smile.de
ProCom International LTD		Kisdorfer Weg 3b	24568 Kaltenkirchen	040 / 71669763	info@procom-international.com
Quinke Networks (PR, Marketing)		Bei den Mühren 70	20457 Hamburg	040 / 43093949	mail@quinke.com
remote control productions (Producing)		Sendlinger-Tor-Platz 6	80336 München	089 / 210205700	info@r-control.de
Rode + Mathé (Rechtsberatung)		Henriettenweg 4	20259 Hamburg	040 / 43270432	info@rodemathé.de
ROESSLER PR (Marketing, PR, Consulting)		Walter-Leisze-Straße 2	60320 Frankfurt am Main	069 / 514461	communicate@roesslerpr.de
S&H Entertainment Localization (Lokalisation)		Weidenstraße 10a	82110 Germering	089 / 80076290	stefan@shentloc.com
SCC (Crossmedia)		Orleansstraße 5a	81669 München	089 / 33094660	info@scf-feld.de
Schanz International (Consulting)		Bgm.-Überhettinger-Straße 2a	67146 Deidesheim	06326 / 6010	info@schanzgames.com
Schulte Riesenkampff (Rechtsberatung)		An der Hauptwache 7	60313 Frankfurt am Main	069 / 900266	schulte@schulte-lawyers.de
Selected Minds (Personalberatung)		Walther-Rathenau-Straße 16	64521 Groß-Gerau	06152 / 1871830	info@selected-minds.de
Serious Games Conference	key player	Expo Plaza 1	30539 Hannover	0511 / 1234560	sgc@nordmedia.de
Sputnic Consulting		Ehrenbergerstr. 11	98693 Ilmenau	03677 / 7996990	media@sputnic-consulting.com
Station54 Community Management		Pankstraße 8-10	13127 Berlin	030 / 44716415	info@station54.de
swordfish PR (PR, Marketing)		Habsburgerplatz 2	80801 München	089 / 96160840	info@swordfish-pr.de
Translation-Taxi		Beusselstraße 28	10553 Berlin	030 / 28371428	info@translation-taxi.com
Turtle Entertainment (eSport)		Siegburger Straße 189	50679 Köln	0221 / 8804490	info@turtle-entertainment.de
Two Pi Team		Gerresheimer Straße 9	40721 Hilden	02103 / 9411914	info@two-pi-team.de
Visibility Communications (PR, Text)		Wichertstraße 16/17	10439 Berlin	030 / 58859941	hi@visi.bi
waza!		Sonnenallee 70	12045 Berlin	0178 / 5223697	jannot@wazaservices.de
FREELANCER BUSINESS SERVICE					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Ahmet Isciturk	key player	Kernstraße 37	90429 Nürnberg	0911 / 27472885	texte@me.com
Anja Weilmann		Kramers Kamp 1a	48291 Telgte	02504 / 7399601	kontakt@spielworte.de
Anne-Petra Lellwitz (Marktforschung)		Fichtestrasse 65	63303 Dreieich	0160 / 2014364	info@apl-mafo.de
Arwed-Ralf Grenzbach (Consulting)		Friedrich-Ebert Anlage 18	60325 Frankfurt am Main	0171 / 6486720	arg@conflutainment.com
Christiane Gehrke		Friedrich-Wilhelm-Straße 15	12103 Berlin	030 / 72010080	cg@tom-putzki-consulting.com
Elisabetta Corapi				0176 / 99763437	info@corapi-translations.com
Falko Löffler (Autor)		Lindenstraße 8a	36355 Grebenhain-Ilbeshausen	06643 / 918577	mail@falkoeffler.de

FREELANCER BUSINESS SERVICE					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Kerstin Fricke	key player	Wissmannstraße 9	12049 Berlin	030 / 20059246	info@kf-uebersetzungen.de
Marc Huppke (Projektmanagement)		Frankfurter Straße 5	64521 Gross-Gerau	06152 / 9614432	marc@make-projects.com
Marc Oberhäuser (Consulting)		Berzbuier Strasse 98	52355 Düren	02421 / 51547	marc.oberhauser@freelancer-games.com
Michael Bhatti (Entwicklung, Consulting)		Bismarckstraße 24	49076 Osnabrück	0541 / 1810660	michael@bhatti.de
Nicole Klinge (PR, Marketing, Events)		Schopenhauerstraße 101	80809 München	0173 / 7872013	nk@gconsultnet.de
Stefan Köhler (Lektor / Autor)		Am Salgenteich 34	38259 Salzgitter	0176 / 22512273	stefanclermenskoehler@web.de
TECHNICAL SERVICE					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Akamai Technologies		Betastraße 10B	85774 Unterföhring	089 / 94006308	contact-germany@akamai.com
Augenpulver (Grafik)		Dotzheimerstraße 164, 4.0G	65197 Wiesbaden	0611 / 94574340	info@augenpulver-design.de
Augmented Minds		Fasangartenstraße 134	81549 München	089 / 69386904	info@augmented-minds.com
BiteTheBytes (Middleware)		St. Gallerer Straße 9	36039 Fulda	0661 / 2006899	info@bitethebytes.com
B.TON Medien (Audio, Musik)		Luise-Ullrich-Straße 4	82031 Grünwald	089 / 189425490	mix@bton.de
Centroid (Motion Capturing)		Schlesische Straße 27	10997 Berlin	030 / 69807474	sven.bergmann@centroidanimation.com
Chromatrix (Mobile Gaming)		Kalkofenstraße 2	72411 Bodelshausen	07471 / 740828	info@chromatrix.com
Codecult (3D-Engine)		Kennastraße 21a	44866 Bochum	02327 / 35474	info@codecult.com
Contentainer (Leveldesign)		Eichelkopfstraße 17	63584 Gründau	06058 / 9178150	info@contentainer.de
Creature Factory		Graefestraße 33	10967 Berlin	030 / 26301330	info@creature-factory.com
Crytek (3D-Engine)	key player	Grüneburgweg 16-18	60322 Frankfurt am Main	069 / 21977660	cryengine@crytek.com
definitiv:ton (Audio)		Mozartstraße 35-37	50674 Köln	0221 / 8234220	mhoechsmann@definitiv-ton.de
Die Kolonie (Art, Design)		Adelheidstraße 82	65185 Wiesbaden	0611 / 5058765	contact@diekolonie.com
Doublesmith (Art/Animation)		Ranzonigasse 3/3	A-3100 St. Pölten	+43 / 699 / 19447880	mail@doublesmith.com
Dutyfarm		Oranienstraße 6	10997 Berlin	030 / 30368430	info@dutyfarm.com
Dynamedion (Sounddesign)		Barbarossaring 8	55118 Mainz	06131 / 547895	info@dynamedion.com
DynamicHead (Unity Services)		Hauptstraße 6	55283 Nierstein Schwabsburg	061 / 33572494	timo.weisse@dynamichead.com
eurosimtec		Merowingerplatz 1	40225 Düsseldorf	0211 / 3018560	info@eurosimtec.de
Exit Games (Network Engine)		Hongkongstraße 7	20457 Hamburg	040 / 4135960	business@exitgames.com
Eye Rock Media (Design, Animation)		Essenerstraße 3	46047 Oberhausen	0208 / 82858923	contact@eyerock.de
fadeout (Audio Design)		Austraße 6	CH-4153 Reinach	+41 / 7 / 92793244	info@fadeout.ch
fatfoogoo		Mariahilferstraße 50	A-1070 Wien	+43 / 1 / 23622970	office@fatfoogoo.com
Flow Studios		Gustav-Meyer-Allee 25	13355 Berlin	030 / 80929203	info@flow-studios.eu
Games Quality (Quality Assurance)		Zum Wasserwerk 15	15537 Erkner	03362 / 885688	info@games-quality.com
Global Producer (Production)		Hauptstraße 172	51143 Köln	02203 / 97722400	getintouch@global-producer.com
Glare Studios (Grafik)		Mansteinstraße 18	20253 Hamburg	02307 / 4388404	contact@glarestudios.de
Goal Games		Feurigstraße 54	10827 Berlin	030 / 78957102	info@goal-games.de
GreenMamba-Studios (Animation)		Aloysstraße 7	48249 Dülmen	02594 / 7928217	info@greenmamba-studios.de
Hans HiScore (Audio)		Friedrich-Ebert-Anlage 11-13	60327 Frankfurt	0151 / 11616133	info@hans-hiscore.de
Havok	key player	Arbachtalstraße 6	72800 Eningen	0712 / 1986993	info@havok.com
Intulo (3D, Animation)		Lindholz 89	31139 Hildesheim	05121 / 2944022	info@intulo.de
Johan Weigel (Musik/Ton/Sound)		Christburger Straße 28	10405 Berlin	0176 / 41446494	johan@sonic-gallery.com
Keuthen		Robert-Koch-Straße 41	55129 Mainz	06131 / 880890	info@keuthen.net
Konsole Labs	key player	Gritznerstraße 42	12163 Berlin	030 / 23634838	info@konsole-labs.com
Lingoona (Lokalisations-Middleware)	key player	Hildastraße 38	68723 Plankstadt	06202 / 4095030	info@lingoona.com
Living Liquid Software		Langer Weg 15	A-6020 Innsbruck	+43 / 0 / 512263535	info@livingliquid.com
metricminds (Grafik)		Rüsselsheimer Straße 22	60326 Frankfurt am Main	069 / 7593380	info@metricminds.com
MobileBits (Mobile Engine)		Lerchenstraße 28	22767 Hamburg	040 / 33429566	info@mobilebits.de
morro images (Grafik)		August-Bebel-Straße 27	14482 Potsdam-Babelsberg	0331 / 9796610	info@morroimages.com
Nevigo (Tools & Middleware)		Lyrenstraße 13	44866 Bochum	02327 / 8369840	info@nevigo.com
Periscope Studio (Audio)		Am Diebsteich 55	22761 Hamburg	040 / 31811767	info@periscopestudio.de
Phenomatics (XNA-Entwicklung, Consulting)		Hafenstrasse 47-51	A-4020 Linz/Donau	+43 / 732 / 90155230	office@phenomatics.com
Polyce (Grafik)		Littenstraße 106/107	10179 Berlin	0178 / 3592560	info@polyce.de
Polygonfabrik (Grafik)		Friesenstraße 21	28203 Bremen		kontakt@polygonfabrik.de
Rabcat Computer Graphics (Grafik)		Fernkornegasse 10	A-1100 Wien	+43 / 1 / 5237425	mail@rabcat.com
Razer (Hardware)		Winterhuder Weg 82	22085 Hamburg	040 / 41929300	
Rocketbox Studios (Grafik)		Leonhardtstraße 10	30175 Hannover	0511 / 8984384	info@rocketbox.de
Spinor (3D-Engine)		Agnes-Pockels-Bogen 1	80992 München	089 / 54043980	spinor@spinor.com
The Light Works (Grafik)		Otto-Hahn-Straße 7	50997 Köln	02236 / 967322	info@thelightworks.com
ToBringAlive (Animation, Design)		Ostmarkstraße 49	48145 Münster	0251 / 392312	angerbauer@tobringalive.com
Tektroop (Support)		Dechenstraße 8	40699 Erkrath		info@tektroop.ch
TÖN & SPOT audiodesign (Sound)		Immanuelkirchstraße 15	10405 Berlin	030 / 92129727	post@ton-und-spot.de
Toygardens Media (Grafik, Video)		Walderseestraße 54	30163 Hannover	0511 / 33659800	akunze@toygardens.com
VCC Perfect Pictures (Grafik, Video)		Doormannsweg 43	20259 Hamburg	040 / 431690	hamburg@vcc.de
Versant (Data Management)		Wiesenkamp 22b	22359 Hamburg	040 / 609900	info@versant.com
Virgin Lands (Grafik, Cinematics)	key player	Gneisenaustraße 10/11	97074 Würzburg	0931 / 8049000	info@virgin-lands.com
xaitment (KI-Middleware)		Fischbacherstraße 92	66287 Quierschied	06897 / 600800	info@xaitment.com
Z-Ground Illustration (Grafik)		Oeltzenstraße 17	30169 Hannover	0511 / 1696959	jz@z-ground.com
zuukal (Audio, Lokalisation)		Christian-Pless-Straße 11-13	63069 Offenbach am Main	069 / 2475700	mail@zuuka.de
FREELANCER TECHNICAL SERVICE					
Firma		Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Alexander N. Ostermann (Animator)		Breite Strasse 45	14199 Berlin	0176 / 10365759	nico@nico-ostermann.com
Andreas Adler (Audio)		Wernigeröder Weg 18	30419 Hannover	0511 / 3889163	andreas@adler-audiopictures.de
Calined (Audio)		Lange Straße 22	58089 Hagen	01577 / 9508456	business@calined.com
Daniel Pharos (Audio)		Golddistelanger 15	80937 München	0179 / 5210100	daniel@knightsoundsoundtrack.com
Grafikbüro Werner (Grafik)		Beethovenstraße 40a	48317 Drensteinfurt	02508 / 985801	info@grafikbuero-werner.de
Johan Weigel (Musik / Ton / Sound)		Christburger Straße 28	10405 Berlin	0176 / 41446494	johan@sonic-gallery.com
Sound42 / Lukas Hasitschka		Kulmgasse 38/8	1170 Wien	+43 / 6641059157	lukas@sound42.com
Maurice T. Anderson (Sounddesign)		Klosterstraße 3	63486 Bruchköbel	0176 / 26228136	contact@maurice-t-anderson.com
Max Schulz					info@xoco1.com
Olaf Bartsch (Sounddesign)		Sültstraße 62	10409 Berlin	0170 / 7743432	info@olafbartsch.com
Sabrina Heuer (Audio Production)		Bachemer Straße 191	50935 Köln	0163 / 8604004	sabrina@froschtatze.de

Ein Tag bei ...

BLACK FOREST GAMES



Unser Logo-Wolfsrudel heult Besuchern beim Betreten unseres Reviers entgegen.



Korrekt, Grün ist unsere Lieblingsfarbe.



Nach der Pause geht's ab ins Meeting.



Pause auf »Black Forest«-Art.



Wir sind immer dabei, die Hand-Auge-Koordination zu trainieren.



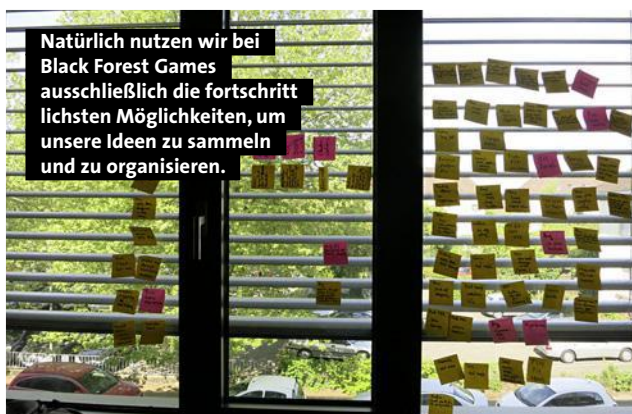
Trotz misstrauischer Blicke darf man sich nie vom Testen der eigenen Arbeit abhalten lassen. Das muss man machen. Ehrlich.



Der Großteil ist da natürlich pflichtbewusster.



Hier werden gerade die nächsten Schritte für »Dieselstörers« geplant.



Natürlich nutzen wir bei Black Forest Games ausschließlich die fortschrittlichsten Möglichkeiten, um unsere Ideen zu sammeln und zu organisieren.



Das Wolfsrudel von Black Forest Games macht Offenburg ein Stück grüner. Und Dank im Büro integrierter Rutsche verliert das Team vor lauter Arbeit den Spaß auch nie aus den Augen.



Kleine Waffen, große Waffen ... Moment, vergessen Sie das mit den kleinen Knarren gleich wieder.



Das Wolfsrudel bei der Arbeit.



Zum Spielen ist man einfach nie zu alt. Zum Rutschen auch nicht.



Nur unsere Eule versteht den Sinn der Rutsche nicht so ganz.



Das sieht leichter aus als es ist. Ehrlich.



Gerne blicken wir zum Schluss noch einmal auf vergangene Erfolge zurück ...



... und bedanken uns herzlich für den Besuch bei Black Forest Games!

MAKING GAMES VORSCHAU

05/2014
MAKING GAMES
ERSCHEINT AM 11.08.2014

Making of ???

In mittlerweile guter alter Tradition werden wir wie jedes Jahr zur gamescom ein Making of zu einem aus unserer Sicht besonders spannenden Spiel am Start haben. Wie immer wird es um ein Spiel gehen, das sich noch in der Entwicklung befindet. Und natürlich wird das Entwicklerteam sein Projekt auch wieder aus den unterschiedlichsten Perspektiven beleuchten. So viel können wir schon mal verraten. Um welches es gehen wird? Tja, da müssen Sie sich leider noch circa zwei Monate gedulden.

Weitere Themen

Bewegungssteuerung mit Kinect 2.0

Was alles mit der neuen Kamera-Generation möglich ist und wie man seine Entwicklungsarbeit an die neue Hardware anpassen muss.

Lords of the Fallen

Das Finale der Fledge-Engine-Serie von Deck13 dreht sich um das CPU- und GPU-basierte Partikelsystem der Engine sowie prototypische Techniken und Systeme.

Interaktives Sounddesign

Filippo Beck Peccoz berichtet von seiner bisher umfangreichsten Arbeit, dem Sounddesign zum Indie-Titel »The Last Tinker: City of Colours«.

IMPRESSUM

Chefredakteur

Heiko Klinge
(verantwortlich,
Anschrift der Red.)

Redaktion

Yassin Chakhchoukh
Patricia Geiger (Trainee)
Ann-Kathrin Kuhls
(Praktikantin)
Markus Schwerdtel

Layout & Design

Sigrun Rüb (ltd.)
Manfred Aumaier
Alexander Wagner
Eva Zechmeister
Anita Blockinger
(Freelancer)
Lektorat Freelancer
Marion Schneider

Freie Mitarbeiter in dieser Ausgabe (Redaktion)

Richard Bartle, Benjamin Glatzel, Jeff Jensen, Klaas Kersting,
Tulay McNally, Nils Plachke, Burkhard Rathelser, David Reinig,
Christian Schmidt, Hans-Ulrich Sesselmann, Thorsten Unger

Wir danken den Interview-Partnern:

James Golding, David Helgason, Andrew Sheppard

Assistenz: Isa Stamp, Anita Thiel, Annie Weissenberger

KONTAKT

Anzeigenanfragen:

Nicole Klinge, 089 / 360 86-680
nklinge@idg-consultant.de

Artikel- & Themenvorschläge:

editor@makinggames.de

Recruiting- & Projektanfragen:

projects@makinggames.de

Adressänderungen Abo:

service@makinggames.de

Änderungen für Firmenregister:

info@makinggames.de

Betreff: Firmenregister

Digitale Anzeigenannahme Print:

Manfred Aumaier, 089 / 360 86-602
anzeigendisoprint@gamestar.de

Digitale Anzeigenannahme Online:

banner@idg.de

Anzeigenpreise:

www.idg-eas.de

HAFTUNG & CO.

© Copyright IDG Entertainment Media GmbH

Haftung: Für die Richtigkeit von Veröffentlichungen können Redaktion und Verlag trotz Prüfung nicht haften. Die Veröffentlichungen in Making Games erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Auch werden Warennamen ohne Gewährleistung einer freien Anwendung benutzt.

KUNDENSERVICE

Abonnement und Einzelbestellung:

Making Games Kundenservice
ZENIT Pressevertrieb GmbH
Postfach 81 05 80, 70522 Stuttgart
Tel.: 0711 / 7252-275
Fax: 0711 / 7252-377

Österreich: Tel.: 01 / 219 55 60

Schweiz: Tel.: 071 / 314 06-15

E-Mail: service@makinggames.de

Web: www.makinggames.de/shop

Bezugspreise:

Einzelpreis: 6,90 Euro
Jahresabo: 35,40 Euro
(beinhaltet 6 Ausgaben)
Auslandspreise u. Mengenrabatt auf Anfrage
Zahlungsmöglichkeit:
Postbank Stuttgart, BLZ 600 100 70
Konto-Nr.: 311 704

So erreichen Sie die Redaktion:

IDG Entertainment Media GmbH
Redaktion Making Games
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-660, Fax: 089 / 360 86-652

VERTRIEB HANDELSAUFLAGE

MZV GmbH & Co. KG
Ohmstraße 1, 85716 Unterschleißheim
Telefon: 089 / 319 06-0, Fax: -113
E-Mail: info@mzv.de
Web: www.mzv.de

VERLAG

IDG Entertainment Media GmbH
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-0, Fax: 089 / 360 86-501
www.idgmedia.de
Gründer: Patrick J. McGovern (1937 - 2014)
Geschäftsführer: York von Heimbürg
Verlagsleitung: Frank Maier
Director Sales: Ralf Sattelberger (-730)
Leiter CRM/Marketing: Matthias Weber (-154)
Veröffentlichung gemäß §8, Absatz 3 des Gesetzes
über die Presse vom 08.10.1949: Alleiniger Gesell-
schafter der IDG Entertainment Media GmbH ist die
IDG Communications Media AG, München, die 100%
Tochter der International Data Group Inc., Boston,
USA, ist.

Vorstand:

York von Heimbürg, Keith Arnot, David Hill

Produktionsleitung: Jutta Eckbrecht

Druck und Beilagen:

SDV Direct World GmbH
01159 Dresden

GDC EUROPE RETURNS
AUGUST 11-13, 2014

**making
games**

READERS,

USE CODE **GDCE14MGP**

TO SAVE AN ADDITIONAL 10% ON
ALL ACCESS PASSES!

SAVE 200€ WITH EARLY REGISTRATION BEFORE JULY 16

GDC 'Eu

GAME DEVELOPERS CONFERENCE™ EUROPE

CONGRESS-CENTRUM OST KOELNMESSE · COLOGNE, GERMANY

AUGUST 11-13, 2014 · EXPO: AUGUST 11-12, 2014

GDCEUROPE.COM

VAASAN

Senior
Programmer

EOGHAN

QA Technician

More than
120 people from
20 different countries
working together on the
banks of the river Spree.

Creating awesome games
since 1999.

Join Our Team!

**WWW.YAGER.DE
/CAREER**

