
Komplett in FARBE
carsten SEIFERT
SPIELE ENTWICKELN MIT
Unity
3D-GAMES MIT UNITY UND C# FÜR DESKTOP, WEB & MOBILE
DVD
Auf DVD: komplettes Spiel plus Videotutorials
HANSER

Seifert

Spiele entwickeln mit Unity

Hinweis:

Zu diesem Buch gehört eine DVD.
Sollte diese DVD nicht beiliegen, können Sie sie unter fachbuch@hanser.de kostenlos anfordern.

Carsten Seifert

Spiele entwickeln mit Unity

3D-Games mit Unity und C# für Desktop, Web & Mobile

HANSER

**»Der Weltuntergang steht bevor,
aber nicht so, wie Sie denken.
Dieser Krieg jagt nicht alles in die Luft,
sondern schaltet alles ab.«**

Tom DeMarco
Als auf der Welt das Licht ausging

ca. 560 Seiten. Hardcover
ca. € 19,99 [D] / € 20,60 [A] / sFr 28,90
ISBN 978-3-446-43960-3
Erscheint im November 2014

Hier klicken zur Leseprobe

**Sie möchten mehr über Tom DeMarco und seine Bücher erfahren.
Einfach reinklicken unter www.hanser-fachbuch.de/special/demarco**

Der Autor:
Carsten Seifert, Süderbrarup
www.hummelwalker.de

Bibliografische Information der Deutschen Nationalbibliothek:

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Lektorat: Sieglinde Schärl
Copy editing: Sandra Gottmann, Münster-Nienberge
Herstellung: Irene Weilhart
Erstellung der Dateien für das Beispiel-Game: Alexej Bodemer, Stuhr, www.alexejbodemer.de
Umschlagdesign: Marc Müller-Bremer, München, www.rebranding.de
Umschlagrealisation: Stephan Rönigk
Gesamtherstellung: Kösel, Krugzell
Printed in Germany

Print-ISBN: 978-3-446-43939-9
E-Book-ISBN: 978-3-446-44129-3

Inhalt

Vorwort

Für viele von uns sind Computerspiele heutzutage allgegenwärtige Wegbegleiter. Egal wo, auf dem Smartphone, dem Tablet oder dem heimischen PC sind sie installiert und täglich in Benutzung. Manchmal dienen sie als Zeitvertreib, bis der nächste Bus kommt, manchmal sind sie aber auch Bestandteil eines intensiven Hobbys.

Aber nicht nur das Spielen kann Spaß machen, auch das Entwickeln dieser Games kann begeistern. Sowohl im Freizeitbereich als auch in der Arbeitswelt wird der Beruf des Spieleentwicklers immer beliebter. Es ist also kein Wunder, dass sich in den letzten Jahren bereits ganze Studiengänge dem Entwickeln von Computerspielen gewidmet haben.

In diesem Buch möchte ich Ihnen Unity, eine weit verbreitete Entwicklungsumgebung für Computerspiele, näherbringen und erläutern, wie Sie mit diesem Werkzeug Spiele selber entwickeln können. Dabei richtet sich das Buch sowohl an Einsteiger, Umsteiger und auch an Spieleentwickler, die mit Unity nun richtig durchstarten möchten.

An dieser Stelle möchte ich mich ganz besonders bei meiner Frau Cornelia bedanken, die mich während des Schreibens so geduldig unterstützt hat und mir jederzeit beim Formulieren und Korrigieren hilfsbereit zur Seite stand.

Auch danke ich ganz herzlich Alexej Bodemer, der für das Beispiel-Game dieses Buches alle 3D-Modelle, Texturen und Musikdateien entworfen und zur Verfügung gestellt hat.

Zudem gilt mein Dank Will Goldstone und Unity Technologies, die mir die bis dato aktuellsten Beta-Versionen zur Verfügung gestellt haben.

Weiter möchte ich Frau Sieglinde Schärl, Kristin Rothe und dem gesamten Hanser-Verlag-Team danken, die mir nicht nur das Schreiben dieses Buches ermöglicht haben, sondern auch jederzeit mit Rat und Tat zur Seite standen.

Nicht zuletzt danke ich auch meiner gesamten Community, die mich während des Schreibens auf meinem Blog und meinen sozialen Kanälen so konstruktiv begleitet hat.

Süderbrarup, August 2014 *Carsten Seifert*

1 Einleitung

Computerspiele gehören heutzutage zu den beliebtesten Freizeitgestaltungen unserer Zeit. Mit Zunahme der Popularität ist aber auch der Anspruch an diese Spiele gestiegen. Während früher ein einzelner Programmierer ausreichte, um alle notwendigen Aufgaben zu erledigen, werden heutzutage anspruchsvolle Computerspiele meist von Teams, bestehend aus Modellierern, Grafikern, Sounddesignern, Level-Designern und natürlich auch Programmierern, umgesetzt.

Um den stetig wachsenden Ansprüchen zu genügen, sind aber auch die Werkzeuge der Entwickler ständig mitgewachsen. Eines dieser Werkzeuge ist Unity. Unity ist eine Spieleentwicklungsumgebung für Windows- und Mac OS X-Systeme und wird von der aus Dänemark stammenden Firma Unity Technologies entwickelt. Mit ihr können Sie sowohl interaktive 3D- als auch 2D-Inhalte erstellen. Ich spreche deshalb von Inhalten und nicht nur von Spielen, weil Unity zwar eigentlich für die Entwicklung von 3D-Spielen gedacht war, mittlerweile aber auch immer häufiger Anwendung in anderen Bereichen findet. So wird es beispielsweise für Architekturvisualisierungen genutzt, im E-Learning-Bereich eingesetzt oder in der Digital-Signage-Branche für das Erstellen digitaler Werbe- und Informationssysteme genommen.

Da Unity ursprünglich für die Entwicklung von 3D-Spielen konzipiert wurde, lautet die Internet-Adresse der Firma *unity3d.com*. Dies ist der Grund, weshalb die Entwicklungssoftware auch gerne mal „Unity3D“ genannt wird, was aber eben nicht ganz korrekt ist.

1.1 Multiplattform-Publishing

Eine besondere Stärke von Unity ist die Unterstützung von Multiplattform-Publishing. Das bedeutet, dass Sie in Unity ein Spiel einmal entwickeln können, das Sie dann aber für mehrere Plattformen exportieren können. Aktuell werden standardmäßig Windows Desktop-Programme, OS X, Linux, Windows Store Apps, iOS, Android, Windows Phone 8, Blackberry 10, Google Native Client sowie ein eigener Webplayer als Publishing-Format unterstützt.

Der erwähnte Webplayer, auch Unity-PlugIn genannt, ähnelt dabei dem Flash Player und wird im Browser installiert. Mit ihm können Spiele, die mit Unity entwickelt wurden, direkt im Browser ohne große funktionale Einbußen gespielt werden.

Sie können mit Unity auch für die Konsolen XBox, Wii und PlayStation entwickeln. Dafür müssen Sie allerdings extra Lizenzen erwerben, die zum einen nicht günstig und zum anderen nur für Firmen verfügbar sind, die von den jeweiligen Konsolenherstellern auch als Entwickler akzeptiert wurden. Deshalb werde ich die Konsolenentwicklung in diesem Buch auch außen vor lassen und nicht näher beleuchten.

1.2 Das kann Unity (nicht)

Unity bringt eine ganze Reihe an nützlichen Werkzeugen mit, um Spiele und andere 3D-Anwendungen zu entwickeln. So gibt es neben einer ausgeklügelten Physik-Engine auch Tools für Partikeleffekte, zur Landschaftsgestaltung oder auch für Animationen. Außerdem wird Unity mit einer extra angepassten Version der Softwareentwicklungsumgebung MonoDevelop ausgeliefert, in der die Programmierung umgesetzt und das Debugging vorgenommen werden kann.

Eines ist Unity allerdings nicht: Es ist keine 3D-Modellierungssoftware. Unity bietet zwar von Haus aus einige 3D-Grundobjekte an, sogenannte Primitives, die für kleinere Aufgaben genutzt werden können, aber für richtige Modellierungsaufgaben sollten Sie auf die dafür entwickelten Spezialtools wie 3ds Max oder das kostenlose Blender zurückgreifen.

1.3 Lizenzmodelle

Die Spieleentwicklungsumgebung gibt es in einer kostenlosen Ausführung (auch Free, Indie oder Unity Basic genannt) und in einer kostenpflichtigen Edition. Die kostenlose Version unterstützt bereits alle oben genannten Zielplattformen und erlaubt eine komplette Spieleentwicklung bis hin zum fertigen Spiel. Die kostenpflichtige Version, die auch Unity Pro genannt wird, beinhaltet dann noch weitere Funktionen, um das Spiel grafisch weiter aufzubereiten, die Performance zu optimieren oder Zusatzfunktionen für Animationen und künstliche Intelligenz. Allerdings sind für bestimmte Plattformen wie iOS, Android und Blackberry 10 noch weitere Lizenzen notwendig, um die zusätzlichen Pro-Features auch dort zu erhalten.

Da sowohl Unity Pro als auch die Indie-Version bzw. Unity Basic kommerziell genutzt werden dürfen, gibt es beim Einsatz der kostenlosen Version die Vorgabe, dass nur Firmen bzw. Entwickler diese einsetzen dürfen, die nicht mehr als 100 000 US-Dollar Umsatz in einem Geschäftsjahr machen.

1.4 Aufbau und Ziel des Buches

Mit diesem Buch werden Sie lernen, auf Basis von Unity eigene 3D-Spiele zu entwickeln. Sie werden sich mit den unterschiedlichen Tools der Game Engine vertraut machen und die Skript-Programmierung erlernen. Dabei steht aber nicht das Ziel im Fokus, wirklich alle Funktionen und Möglichkeiten zu beleuchten, die Unity dem Entwickler anbietet. Vielmehr zeige ich Ihnen, wie die unterschiedlichen Bereiche von Unity funktionieren und miteinander zusammenarbeiten. Denn der Funktionsumfang dieser Spieleentwicklungsumgebung ist mittlerweile so umfangreich geworden, dass es gar nicht mehr möglich ist, alle Tools und deren Möglichkeiten bis ins letzte Detail in einem einzigen Buch ausführlich zu behandeln. Daher liegt der Schwerpunkt auf den Kernfunktionen, die in der kostenlosen Unity-Version für die 3D-Spieleentwicklung bereitgestellt werden, ergänzt um Anwendungsbeispiele und Praxistipps.

Möchten Sie weiter in die Tiefe eines speziellen Tools gehen oder mehr Informationen zu bestimmten Scripting-Klassen erhalten, empfehle ich Ihnen die mit Unity mitgelieferten Hilfe-Dokumente, die Sie über das Help-Menü erreichen. Dort finden Sie sowohl ein ausführliches Manual über die in Unity integrierten Tools sowie eine *Scripting-Referenz* über alle Unity-Klassen und deren Möglichkeiten. Letztere finden Sie auch über die Hilfe von MonoDevelop, der mitgelieferten Programmierumgebung.

Spieleentwicklung wird häufig auch als Spieleprogrammierung bezeichnet, auch wenn viele Aufgaben in der Spieleentwicklung heutzutage nicht mehr programmiert, sondern mithilfe von Tools erledigt werden. Nichtsdestotrotz ist der Programmieranteil bei der Entwicklung eines Spiels doch immer noch sehr hoch. Deshalb wird auch das Buch zunächst mit zwei größeren programmierbezogenen Kapiteln beginnen, wo es unter anderem auch um die Grundlagen der Programmierung geht. Erst danach werden wir in die 3D-Welt von Unity eintauchen und die verschiedenen Werkzeuge behandeln. Der Aufbau ist deshalb so gewählt, weil ich in den folgenden Kapiteln immer wieder kleinere Skript-Beispiele zeige, die Einsatzmöglichkeiten in der Praxis demonstrieren.

Am Ende des Buches werden wir schließlich in einem etwas größeren Kapitel ein Beispiel-Game entwickeln, das viele der behandelten Themen noch einmal aufgreift und die gesamten Zusammenhänge in der Praxis zeigt.

Die DVD zum Buch

Dem Buch liegt eine DVD bei, auf der sich Folgendes befindet:

- Beispiel-Game (Dungeon Crawler) mit allen dazugehörigen Ressourcen
- Anwendungsbeispiel für eine Auto-Steuerung inklusive eines 3D-Modells mit zusätzlichen Skripten
- Beispiel für eine Sprite-Animation
- Vorlage für ein einfaches Übungsprojekt
- ergänzende Video-Tutorials, in denen die Verwendung spezieller Werkzeuge und Verfahren in der Praxis demonstriert wird.

Im Buch werde ich Hinweiskästen, wie den obigen, einsetzen, um Hinweise und Tipps zu geben. Je nach Typ des Hinweises werden diese mit unterschiedlichen Icons ausgezeichnet.

Hinweise zu Inhalten auf der DVD

Hinweise zu weiterführenden Inhalten im Internet

Praxistipps

Allgemeine Hinweise

1.5 Weiterentwicklung von Unity

Die Spieleindustrie gehört zu den Branchen, die sich aktuell am schnellsten verändern. Kein Wunder also, dass auch Unity ständig weiterentwickelt wird und neue Funktionen erhält. Sollten Sie Unterschiede zwischen dem Buch und Ihrer Unity-Version erkennen, wird dies sicher der ständigen Weiterentwicklung von Unity geschuldet sein.

Aber nicht nur der Funktionsumfang wird ständig weiterentwickelt, auch das Lizenzmodell von Unity ist nicht von Veränderungen ausgeschlossen. Vielleicht können Sie schon Funktionen mit Unity Basic nutzen, die in diesem Buch noch als Pro-Feature gelten. Ein gutes Beispiel hierfür sind die Berechtigungen für die Mobile-Entwicklung. Noch vor Kurzem hätten Sie bereits für die Basic-Mobile-Entwicklung extra Lizenzen kaufen müssen. Diese Berechtigungen sind jetzt bereits in der kostenlosen Version integriert. Und auch das Add-On für das Pathfinding (siehe Kapitel „Künstliche Intelligenz“) war anfangs ein Pro-Feature, heutzutage ist es fast komplett in der Indie-Version enthalten.

1.6 Online-Zusatzmaterial

Aufgrund der kontinuierlichen Weiterentwicklung von Unity werde ich Ihnen in einem passwortgeschützten Bereich meiner Website Zusatzmaterialien bereitstellen, die den Inhalt dieses Buches ergänzen und auch aktuell halten sollen.

Passwortgeschützter Bereich für Zusatzmaterialien

URL: *http://www.hummelwalker.de/buch-zusatzmaterial/*

Passwort: **uN1TyZuS4TzMaT3RiAl**

2 Grundlagen

In diesem Kapitel werden Sie die Oberfläche von Unity sowie deren grundlegende Bedienung kennenlernen. Wir werden auf die Struktur eines Unity-Projektes eingehen und die grundlegenden Prinzipien von Unity behandeln.

2.1 Installation

Sollten Sie noch keine Installationsdatei haben, besuchen Sie zunächst die Seite *http://www.unity3d.com* und laden sich dort die aktuelle Unity-Version herunter. Nach dem Download können Sie Unity installieren. Da die Installation eigentlich selbsterklärend ist, sollten Sie lediglich darauf achten, dass Sie eine Komplettinstallation machen und keine Teile davon ausnehmen. Profis können natürlich selber bewerten, was sie benötigen, Anfängern würde ich aber immer eine Komplettinstallation empfehlen.

Wenn Sie die Installation abgeschlossen haben und Unity das erste Mal starten, wird sich der *Project Wizard* von Unity zeigen. Über diesen können Sie ein existierendes Projekt öffnen oder ein neues Projekt erstellen. Mehr zum Öffnen und Erstellen eines Projektes erfahren Sie im Abschnitt „Das Unity-Projekt“. Für den ersten Start können Sie einfach die Standardeinstellungen übernehmen und über **Create** ein neues Projekt erstellen.

2.2 Oberfläche

Die Oberfläche von Unity besteht aus mehreren frei anpassbaren Fenstern (auch Tabs genannt), die sich innerhalb von Unity übereinander und nebeneinander andocken sowie auch außerhalb des Hauptfensters verschieben lassen.

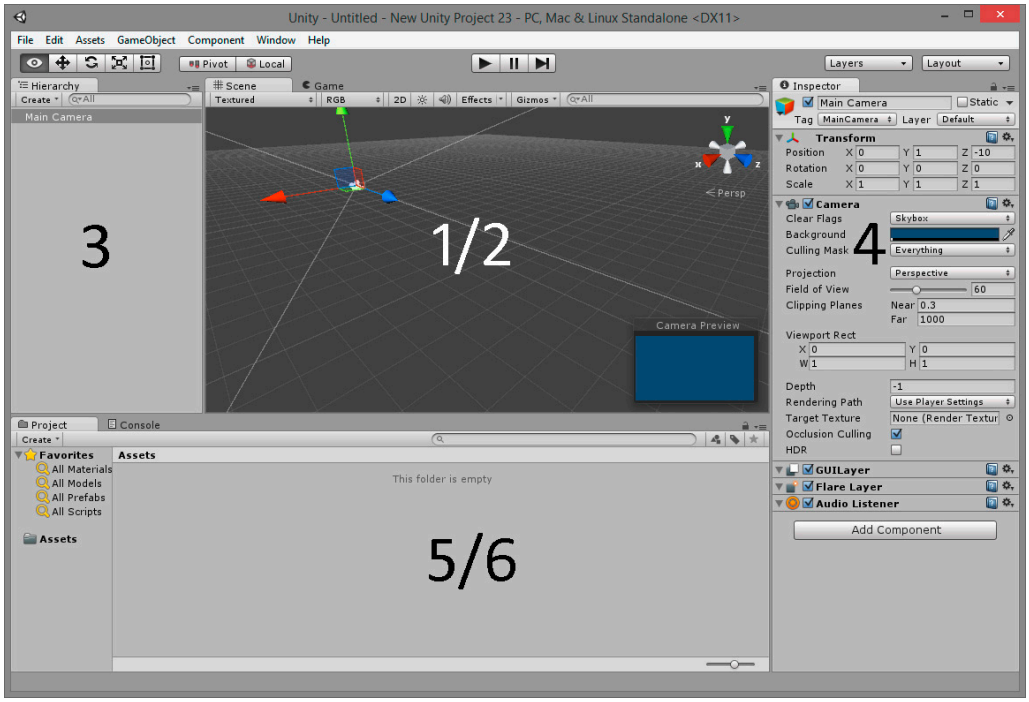

Bild 2.1 Default-Ansicht von Unity

Die Fenster (Tabs) besitzen zwar unterschiedliche Aufgaben, gehören funktional aber alle zusammen und werden deshalb auch kontinuierlich untereinander synchronisiert.

1. **Scene View** dient dem interaktiven Gestalten von Szenen und 3D-Welten.
2. **Game View** dient als Vorschau des fertigen Spiels. Dieses Fenster wird in Bild 2.1 von der *Scene View* verdeckt, wird aber von Unity automatisch nach vorne gebracht, wenn das Spiel gestartet wird.
3. **Hierarchy** zeigt alle in der Szene existierenden Objekte (*GameObjects*) in deren Hierarchiestruktur an.
4. **Inspector** zeigt alle Komponenten und öffentlichen Parameter des aktuell selektierten *GameObjects* an.
5. **Project Browser** dient dem Anzeigen und Verwalten aller digitalen Inhalte (auch *Assets* genannt), die zu dem Projekt gehören.
6. **Console** dient dem Anzeigen von Fehler- und Hinweismeldungen. Dieses Fenster befindet sich in der Default-Ansicht hinter dem *Project Browser*. Um die Meldungen zu sehen, müssen Sie auf den Reiter des Fensters klicken.

Abgesehen von der *Game View* und dem *Console-Tab* können Sie den Tabs nicht nur Informationen entnehmen, sondern auch die Objekte verändern.

Sie können die Tab-Anordnungen in Unity jederzeit speichern und auch zurücksetzen. Hierfür stellt Unity im oberen Hauptmenü über **Window/Layouts** entsprechende Funktionen und Standardanordnungen zur Verfügung. Als Schnellzugriff dient hier das *Layouts-Dropdown-Menü*, das Sie ganz rechts im oberen Bereich von Unity finden.

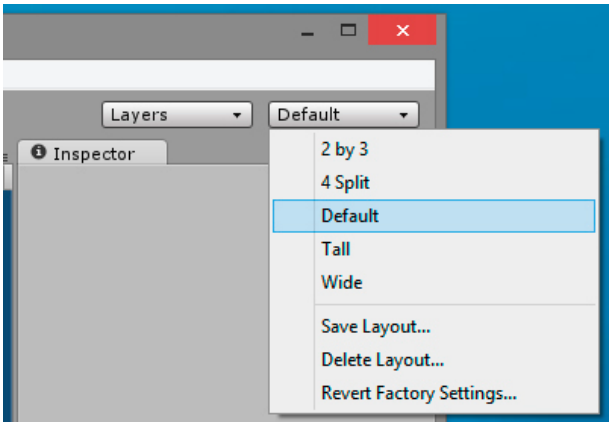

Bild 2.2
Schnellzugriff auf die Layout-Funktionen

Wenn Sie mit **Save Layout** eigene Fenster-Anordnungen speichern, stehen diese Layouts nicht nur in diesem Projekt zur Verfügung, sondern auch in allen anderen Unity-Projekten. Mit **Delete Layout** löschen Sie diese natürlich ebenfalls in allen Projekten. Sollten Sie dabei eine Standardsicht aus Versehen gelöscht haben, können Sie über **Revert Factory Settings** alle Layout-Einstellungen wieder auf die Werkseinstellungen zurücksetzen.

2.2.1 Hauptmenü

Oberhalb aller Subfenster und Toolbars befindet sich das Hauptmenü von Unity. Viele Teile dieses Menüs finden Sie zusätzlich noch einmal als Schnellzugriff-Menüs in anderen Bereichen von Unity wieder, wie z. B. das *Layouts-Drop-down-Menü* (siehe oben).

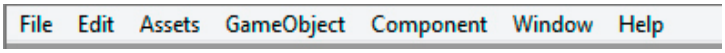

Bild 2.3 Hauptmenü

Hinter diesen Hauptmenüpunkten verbergen sich folgende Funktionen:

- **File** beinhaltet alle Funktionen, die sich mit dem Erstellen und Speichern von Dateien auf Projekt- und Szene-Ebene beschäftigen.
- **Edit** besitzt vor allem Einstellungen zum Verändern von Daten.
- **Asset** beschäftigt sich mit dem Verwalten und Erstellen neuer *Assets*, z. B. von Skripten und Materialien.
- **GameObject** beschäftigt sich vor allem mit dem Erstellen verschiedener, vorkonfigurierter *GameObjects*.
- **Component** bietet alle Standardkomponenten von Unity an.
- **Window** bietet vor allem weitere Fenster/Tabs an, die Sie anzeigen können.
- **Help** besitzt hauptsächlich Quellen und Links zu weiterführenden Informationen für Unity. Hier gelangen Sie auch zum „Unity Manual“ und zu der „Scripting Reference“, wo Sie viele weiterführende Informationen finden.

2.2.2 Scene View

Das Fenster mit der Überschrift „Scene“ wird als *Scene View* bezeichnet und dient dem interaktiven Positionieren und Verändern von Objekten innerhalb einer Szene bzw. eines Levels. Sie können hier neue Objekte hinzufügen, diese verschieben, rotieren und auch skalieren. Ein wichtiges Werkzeug hierfür sind die *Transform-Tools*, die Sie in der Toolbar direkt unter dem Hauptmenü oben links finden. Über Maus und Tastatur können Sie zudem durch die Szene navigieren (siehe Abschnitt 2.2.2.1, „Navigieren in der Scene View“).

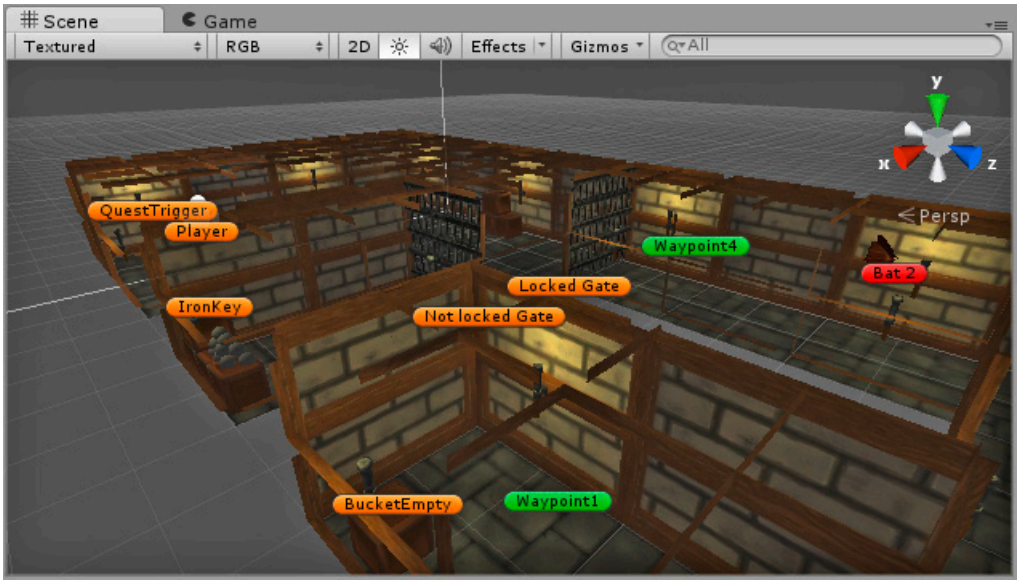

Bild 2.4 Scene View

Möchten Sie ein neues *GameObject* in der *Scene View* platzieren, können Sie einfach ein *GameObject* aus dem *Project Browser* in die *Scene View* hineinziehen und fallen lassen. Anschließend können Sie es noch verschieben und mit den erwähnten *Transform-Tools* modifizieren (siehe Abschnitt 2.2.4, „Toolbar“).

Am oberen Rand der *Scene View* befindet sich eine schmale *Control Bar*, die Ihnen über mehrere Untermenüs verschiedene Einstellmöglichkeiten bietet (siehe Bild 2.4).

- **Mesh Rendering-Menü** wechselt die Ansicht zwischen verschiedenen Darstellungsarten (Texturiert, Drahtgitter usw.).
- **Texture Rendering-Menü** stellt verschiedene Darstellungsformen der Texturen zur Verfügung (RGB, Alpha …).
- **2D-Button** wechselt zwischen der normalen 3D-Ansicht und einer speziellen Darstellungsform, die besonders für 2D-Games und das GUI-Design geeignet ist. Mehr zu dieser Ansicht erfahren Sie im Abschnitt 2.2.2.2.
- **Beleuchtungs-Button** schaltet die Darstellung der Lichtquellen bzw. deren Auswirkungen in der *Scene View* ein/aus.

- **Sound-Button** schaltet den Ton ein/aus.
- **Effects-Menü** ermöglicht, bestimmte Effekte in der Szene zu aktivieren/deaktivieren (Skybox, Flares …).
- **Gizmos-Menü** schaltet typbezogen Hilfsgrafiken in der *Scene View* ein/aus (Gizmos sind grafische Hilfsmittel, um nicht sichtbare Objekte wie Kameras, Soundquellen etc. darzustellen).
- **Suchmaske**, um Objekte in der Szene zu suchen (dabei wird alles ausgegraut, außer die gefundenen Objekte)

2.2.2.1 Navigieren in der Scene View

Unity bietet unterschiedliche Möglichkeiten, durch die *Scene View* zu navigieren. Manche Funktionen sind hierbei auch auf unterschiedliche Weise zu erreichen. Im Folgenden möchte ich Ihnen die wichtigsten vorstellen.

- **Objekte selektieren** Sie mit der linken Maustaste.
- **Ansicht in der Höhe und Seite verschieben** Sie mit der mittleren Maustaste bzw. durch Drücken auf das Mausrad. Sie können auch über die Pfeiltasten navigieren oder alternativ auch das Verschiebewerkzeug der *Transform-Tools* aktivieren (Shortcut: Q) und dann die linke Maustaste nutzen (siehe *Transform-Tools*). Wenn Sie große Szenen besitzen, können Sie die Geschwindigkeit des Verschiebens über das zusätzliche Drücken der Umsch-Taste (Shift) noch erhöhen.
- **Ansicht in der Tiefe und Seite verschieben** Sie über die Pfeiltasten. Hierbei bewegen Sie sich den Pfeiltasten entsprechend durch die Szene.
- **Ansicht drehen** Sie über die rechte Maustaste. Alternativ können Sie auch die Kombination aus Alt-Taste und der linken Maustaste nutzen.
- **Ansicht zoomen** Sie über das Drehen des Mausrads. Alternativ können Sie auch die Alt-Taste und die rechte Maustaste drücken, während Sie den Mauszeiger bewegen.
- **Objekte fokussieren** Sie, indem Sie ein Objekt in der Scene View oder in der *Hierarchy* selektieren und dann die Taste F drücken. Die Ansicht wird dann so gesetzt, dass das Objekt im Mittelpunkt steht.

2.2.2.2 Scene Gizmo

Oben rechts innerhalb der *Scene View* finden Sie das *Scene Gizmo*, das die Orientierung des Koordinatensystems zur aktuellen Sicht der *Scene View* darstellt. Das *Scene Gizmo* ist dabei wie ein Kompass und zeigt Ihnen, von welcher Seite Sie aktuell die Szene betrachten.

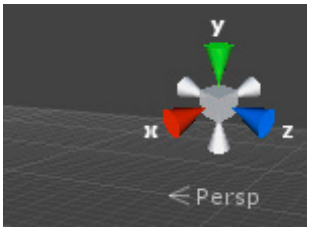

Bild 2.5
Scene Gizmo

Wenn Sie auf die unterschiedlichen Achsen des *Scene Gizmos* klicken, wechselt die Ansicht dabei in die jeweilige Seiten-, Drauf- oder Frontansicht. Ein Klick auf den mittleren Würfel ändert zudem die Ansicht zwischen orthogonal und perspektivisch.

- **Orthogonal** ignoriert Tiefenverhältnisse. So werden beispielsweise zwei gleich große Objekte immer gleich groß dargestellt, egal wie weit diese vom Betrachter entfernt sind.
- **Perspektivisch** stellt Objekte „natürlich" dar und berücksichtigt die Entfernung vom Betrachter.

Unter dem *Scene Gizmo* wird Ihnen in Textform zusätzlich angezeigt, welche Ansicht Sie aktuell gewählt haben. Die drei parallel verlaufenden Linien bedeuten hierbei orthogonal, die auseinandergehenden Linien bedeuten perspektivisch. Ein Klick auf diesen Text wechselt ebenfalls zwischen diesen beiden Ansichtsformen.

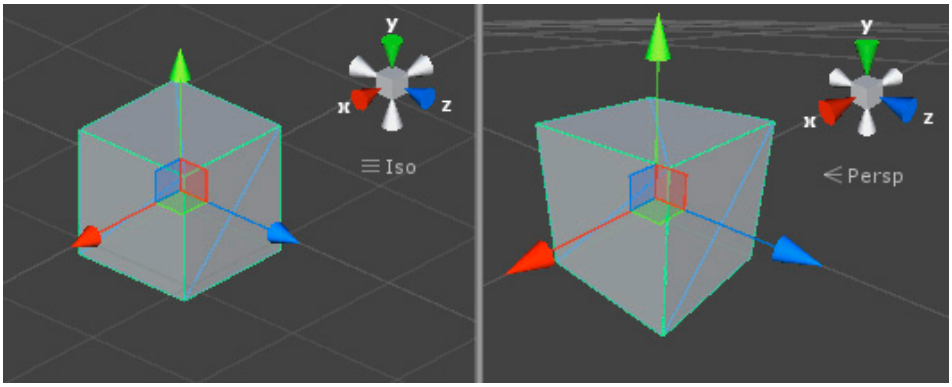

Bild 2.6 Vergleich: Orthogonal (links) vs. Perspektivisch (rechts)

2D-Button

Der *2D-Button*, den Sie in der *Control Bar* der *Scene View* finden, ändert lediglich die Ansicht auf „Orthogonal" und dreht die räumliche Betrachtung so, dass der Nutzer in Richtung der positiven Z-Achse schaut, während die Y-Achse nach oben zeigt. Er ruft also nur eine Standardansicht auf, die sich für 2D-Spiele als praktisch erwiesen hat. Ansonsten nimmt dieser Button keinen Einfluss auf das Spiel.

2.2.3 Game View

Die *Game View* dient dem Testen Ihres Projektes. Sie zeigt das Spiel aus Sicht der Kamera und ermöglicht Ihnen, das Spiel so zu betrachten, wie es später nach der Erstellung aussehen wird. Wenn Sie das Spiel über die Play-Taste starten, können Sie über dieses Fenster Ihr Spiel testen.