



Altova, der Hersteller des bekannten XML-Editors XMLSpy hat die Version 2010 des Altova MissionKits, eines integrierten Pakets bestehend aus XML-, Datenbank- und UML-Tools, auf den Markt gebracht. Version 2010 enthält nun in seiner gesamten Produktlinie 70 der am häufigsten gefragten Features – angefangen von der Unterstützung für neue Technologien wie WSDL 2.0 in XMLSpy und MapForce, JSON in XMLSpy und SysML in UModel, bis zu verbesserten Funktionalitäten für die Arbeit mit XBRL in XMLSpy, einer neuen Stylesheet-Design-Option in StyleVision und vieles mehr. Version 2010 ist dazu auch zu Windows 7 kompatibel.

Der XMLSpy XML-Editor wurde um umfangreiche neue Funktionalitäten erweitert, darunter auch die Unterstützung für WSDL 2.0. Diese Funktion gibt es nun zusätzlich zur Unterstützung für WSDL 1.1. XMLSpy unterstützt die Konvertierung zwischen WSDL 1.1- und WSDL 2.0-Dateien. Dazu kommt eine verbesserte XBRL-Unterstützung, wie ein neuer XBRL-Taxonomieassistenten. Neue Funktionen erleichtern die Arbeit mit JSON. Der neue JSON-Editor gibt dem Benutzer die Flexibilität, JSON-Dateien ähnlich intuitiv wie XML-Dateien zu editieren, einschließlich Konvertierung zwischen XML- und JSON-Dateien.

Version 2010 des UML-Tools UModel wurde durch neue produktivitätssteigernde Features wie unter anderem Unterstützung für SysML (Systems Modeling Language) ergänzt. UModel 2010 unterstützt alle SysML v1.1-Diagrammtypen und -elemente sowie die Codegenerierung in Java, C# und Visual Basic. Außerdem wurden eine neue Skripting-Umgebung und ein Formular-Editor hinzugefügt, damit UModel-Operationen vom UML-Tool aus selbst automatisiert werden können. Des Weiteren bietet UModel 2010 eine neue Option zum Generieren mehrerer Sequenzdiagramme anhand von mit Reverse-Engineering erstelltem Java-, C#- und Visual Basic-Applikationscode.

DatabaseSpy 2010 enthält eine leistungsstarke neue Funktion zum visuellen Vergleichen von Datenbankschemas, mit der sowohl Datenbanken desselben Typs als auch solche unterschiedlichen Typs verglichen werden können (alle gängigen relationalen Datenbanken werden unterstützt).

Info: [www.altova.com](http://www.altova.com)

## Sound und Video im XNA Framework

Am Erfolg eines Spieles ist die **auditive Untermalung** mindestens genauso beteiligt wie **visuelle Effekte**. Mit dem »Microsoft Cross-Platform Audio Creation Tool« des XNA-Game-Studios haben Sie **eines der besten Werkzeuge zur Hand, Ihre Spielwelten mit den passenden Klangwelten zu unterlegen**.

Neben einer realitätsgetreuen Darstellung der virtuellen Welt ist die auditive Untermalung maßgeblich an der so genannten »Immersion« (siehe Kasten) beteiligt. Die Wiedergabe der Audiodaten muss dabei diverse Kriterien erfüllen. Eines der Kriterien ist die Glaubwürdigkeit. Das bedeutet zum Beispiel, dass Objekte in einer dreidimensionalen Welt mit zunehmender Entfernung leiser werden. Ferner muss die Vertonung unmittelbar mit der visuellen Repräsentation einhergehen – kaum etwas stört mehr die Illusion als eine asynchrone Geräuschkulisse.

### IMMERSION

„Immersion“ bezeichnet, salopp gesagt, das Eintauchen in eine virtuelle Welt, wobei die Wahrnehmung der eigenen Person herabgesetzt wird. Beim Spieler entsteht der Eindruck, sich unmittelbar in der virtuellen Welt zu befinden und diese beeinflussen zu können.

Mit dem »Microsoft Cross-Platform Audio Creation Tool« (XACT) sind Sie in der Lage, all diese Kriterien erfüllen zu können. XACT entkoppelt den Audiodesign-Prozess von der Entwicklung, indem Microsoft Ihnen mit diesem Tool erlaubt, die Sounddaten zu bearbeiten. Daraus resultiert ein Paket, das Sie in die Anwendung einbinden. Demzufolge brauchen Entwickler keinen Gedanken mehr an die Konfiguration der Lautstärke oder Effekte zu verschwenden. XACT ist Bestandteil der XNA Game Studio-Installation. Dem DirectX SDK liegt das Tool ebenfalls bei. XACT gestattet sowohl die zweidimensionale als auch die dreidimensionale Audiowiedergabe.

Zu den neuen Features von XACT<sub>3</sub>, das mit dem XNA Game Studio 3.1 eingeführt worden ist, zählt die Dekodierung von xWMA. Dieses Format stützt sich auf das Kompressionsformat »Windows Media Audio« (WMA). Da die Dekodierung eine Softwarelösung ist, geht dies zu Lasten der CPU. Dafür funktioniert die Dekodierung sowohl auf dem Computer wie auch auf der Xbox 360. Microsoft empfiehlt den Einsatz von xWMA insbesondere für Dialoge oder Menüs – in Einsatzgebieten also, in denen Sie locker auf ein paar CPU-Zyklen verzichten können und als Gegenleistung dafür eine höhere Kompression bekommen.

### Ein wenig Terminologie

Bevor es zur Praxis übergeht, bedarf es der Klärung der Terminologie jener Begriffe, die im Zusammenhang mit XACT auftreten. Sowohl in diesem Tool, als auch im

XNA Framework sowie dessen Dokumentation begegnen Sie folgenden Begriffen: »Waves«, »Wave Bank«, »Sound Bank« und »Cue«.

- *Waves* repräsentieren unabhängige Audio-Dateien, die Sie in der Regel im XACT importieren oder direkt in der Anwendung mittels der Sound-Effect-Klasse unabhängig vom XACT abspielen.
- Eine *Wave-Bank* bildet eine Gruppe von Audiodateien, die in einer Datei gruppiert sind.
- Eine *Sound-Bank* ist wiederum eine Gruppe von Wave-Banks und Cues.
- *Cues* erlauben dem Entwickler Sounds abzuspielen, die aus einem oder mehreren Sounds bestehen. Cues referenzieren Sie über eine Sound-Bank.

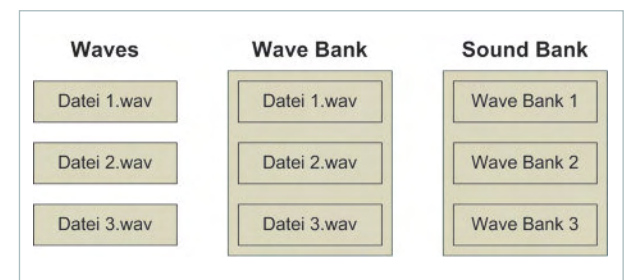


Abb. 1: Die XACT-Terminologie

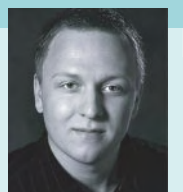
### Zweidimensionale Audiowiedergabe

Mit der zweidimensionalen Audiowiedergabe ist lediglich die einfache Wiedergabe der Sounddaten gemeint, ohne dass das API Positionierungen im dreidimensionalen Raum berücksichtigt. Salopp gesprochen entspricht die zweidimensionale Audiowiedergabe der Ausgabe von Mono- und Stereo-Sounddaten. Der erste Schritt besteht im Anlegen eines neuen XACT-Projekts. Im nächsten Schritt legen Sie eine neue Wave-Bank und eine neue Sound-Bank an. Für Elemente der Sound-Bank nehmen Sie individuelle, auf die Szenerie abgestimmte Konfigurationen der Audiodaten vor. Folglich dürfen mehrere Elemente auf eine Wave-Datei der Wave-Bank verweisen.

Die Menüpunkte »Wave Banks« und »Sound Banks« offerieren jeweils einen Befehl zum Anlegen eines neuen Elements. Alternativ wählen Sie den entsprechenden Befehl im Kontextmenü der Knoten im Projektbaum auf der linken Seite an. Erweitern Sie die Wave-Bank um beliebige Dateien. Das ist wiederum möglich über die Menüleiste oder das Kontextmenü des

### JENS KONEROW

E-Mail:  
[webmaster@jenskonerow.de](mailto:webmaster@jenskonerow.de)



Jens Konerow ist freier Software-Entwickler, mehrfacher Buchautor und Microsoft Most Valuable Professional (MVP) im Bereich DirectX/XNA.

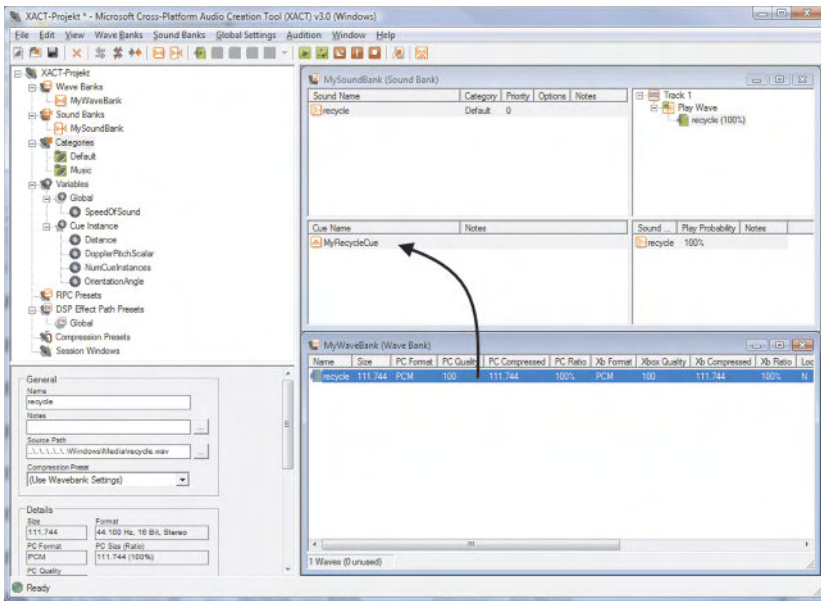


Abb. 2: Generierung eines Cues

Knotens beziehungsweise des MDI-Fensters Wave-Bank.

Die Benutzeroberfläche zur Bearbeitung einer Sound-Bank ist zweigeteilt. In der oberen Hälfte listet das Fenster alle Sounds auf, die der Sound-Bank angehören. Die untere Hälfte zeigt die Cues, die auf Basis der Sounds existieren. Ziehen Sie eine Datei aus der Wave-Bank in den Cue-Bereich der Sound-Bank. XACT übernimmt die Datei als Sound und instanziiert gleichzeitig ein Cue (siehe Abbildung 2).

Das soeben erstellte XACT-Projekt dient als Referenz für die Content-Pipeline, weshalb Sie dem XNA Game Studio-Projekt die neue XAP-Datei als Game-Asset hinzufügen. Den korrekten Content-Importer sowie Content-Processor wählt die Entwicklungsumgebung automatisch. Beide hören auf den Namen »XACT Project – XNA Framework«. Sobald der Quellcode kompiliert wird, stößt die Entwicklungsumgebung die Content-Pipeline an, die, jeweils auch für die Windows- und Xbox 360-Version, mindestens drei Dateien erzeugt:

- **XGS-Datei:** Speichert globale Informationen
- **XWB-Datei:** Enthält die Wave-Banks
- **XSJ-Datei:** Enthält die Sound-Banks

Voraussetzung für die Wiedergabe in einer XNA-Framework-basierten Anwendung ist eine AudioEngine-Instanz. Jene Objekte initialisieren die Sounds und übernehmen die Wiedergabe. Ferner ist jeweils mindestens ein Objekt des Typs WaveBank beziehungsweise SoundBank erforderlich. Die Initialisierungsroutinen platzieren Sie in der Initialize-Methode der Game-Klasse.

```
protected override void Initialize()
{
    m_oAudioEngine = new AudioEngine(
        "Content\\Sounds\\XACT-Projekt.xgs");
    m_oWaveBank = new WaveBank(
        m_oAudioEngine, "Content\\Sounds\\MyWaveBank.xwb");
    m_oSoundBank = new SoundBank(
        m_oAudioEngine, "Content\\Sounds\\MySoundBank.xsb");

    base.Initialize();
}
```

Im Gegensatz zur gewohnten Vorgehensweise bemühen Sie nicht den Content-Manager, um das XACT-Projekt zu laden. Stattdessen übergeben Sie den Pfad der

Content-Pipeline-Elemente direkt dem Konstruktor der jeweiligen Klasse. Beachten Sie, dass an dieser Stelle die Dateierweiterungen angegeben werden müssen.

Die Namensgebung der XGS-Datei leitet sich aus dem XACT-Projektnamen ab, den Sie beim Speichern angegeben haben. Hingegen konfiguriert man die Namen der Wave-Bank und Sound-Bank innerhalb von XACT, indem Sie das jeweilige Objekt auswählen und der Eigenschaft »Windows Build Path« beziehungsweise »Xbox 360 Build Path« einen Dateinamen zuweisen (siehe Abbildung 3, unten).

Die Audio Engine ist eine Pflichtkomponente. Allein schon deshalb, weil die Konstruktoren der

Klassen WaveBank und SoundBank eine solche Instanz als Argument erwarten. Ferner ist ein Aufruf der Update-Methode pro Frame fällig.

```
protected override void Update(GameTime gameTime)
{
    m_oAudioEngine.Update();
    base.Update(gameTime);
}
```

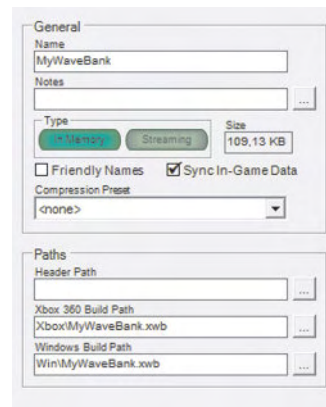


Abb. 3: Vergabe der Dateinamen für Wave- und Sound-Bank

Die Vorbereitungsmaßnahmen sind nun abgeschlossen. Jetzt steht lediglich die Wiedergabe der Audioquellen aus. Am schnellsten kommen Sie mit der PlayCue-Methode eines SoundBank-Objekts zum Ziel. Dessen Methode erwartet lediglich den Cue-Namen. Greifen Sie auf diese Variante zurück, wenn Sie keinen Einfluss auf die Wiedergabe ausüben möchten. Mehr Kontrollmöglichkeiten bietet die Cue-Klasse. Objekte dieses Typs liefert die Methode SoundBank.GetCue. Anschließend üben die Methoden Play, Pause, Resume und Stop Einfluss auf die Eigenschaften IsPaused, IsPlaying, IsStopped und IsStopping aus. Mit Ausnahme der Stop-Methode sind alle oben aufgezählten Methoden parameterlos. Stop erwartet eine Konstante der AudioStop-Options-Enumeration. Diese Enumeration definiert zwei Konstanten: AsAuthored und Immediate. AsAuthored bewirkt, dass die Audio Engine beispielsweise vordefinierte Übergänge zwischen zwei Sounds berücksichtigt. Immediate hingegen hält die Wiedergabe unmittelbar an.

**»In Memory« versus »Streaming«**

Nachdem Sie im XACT eine Wave-Bank ausgewählt haben, bietet Ihnen die Applikation zwei Typen zur Auswahl an: »In Memory« oder »Streaming« (siehe Abbildung 3, Mitte).

Mit der Roadmap 2010 für seine tool-basierte Dienstleistung „The Porting Project“ stellt das Beratungs- und Softwarehaus fecher neue Optionen zur Portierung von Gupta-Team-Developer-Anwendungen auf das .NET-Framework vor. Wichtigste Neuerung bilden die PPJ Services, eine Technologie zur Migration in serviceorientierte Architekturen (SOA), Rich Internet Applications (RIA) oder andere, auf Cloud-Computing optimierte Architekturen.

Seit Beginn des „Porting Projects“ im Jahr 2006 hat fecher gut 130 Altanwendungen für Kunden von der 4GL-Umgebung Gupta Team Developer auf VB.NET oder C# umgestellt und damit in die .NET-Umgebung von Microsoft portiert. Unter Zuhilfenahme der eigens entwickelten .NET-Klassenbibliothek PPJ-Framework entstanden dabei aus ehemals monolithischen Anwendungen moderne .NET-Applikationen, die in Visual Studio gepflegt und nach allen Regeln modernen Softwareengineerings weiterentwickelt werden können.

PPJ Services erweitern das PPJ Framework um die Möglichkeit zur Partitionierung in mehrschichtige Architekturen. So können Anwendungsteile wie SQL-Zugriffe, Reporting oder beliebige andere Objekte vom Desktop auf einen Server, ins Rechenzentrum oder zu einem Cloud-Computing-Anbieter verlegt werden. Das enthaltene Rich Internet Application (RIA) Framework erlaubt es sogar, die gesamte Anwendung auf eine Server- oder Cloud-Plattform zu verlegen, so dass auf dem Client nur noch ein Browser benötigt wird. Als technische Plattform kann frei zwischen AJAX und Silverlight gewählt werden.

Info: [www.fecher.eu](http://www.fecher.eu)

-Anzeige-

**ppedv-Quicky**

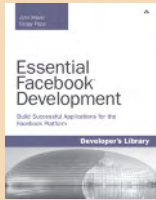
Admin Job automatisieren mit Powershell

„dosbox meets systemcenter“  
Lernen Sie, wie man sich mit der Powershell seine Systeme dienstbar macht. Und das OHNE Programmiergrundkenntnisse.  
Powershell 2 aus Windows 7 und Server 2008 R2 sind dabei inklusive.

**Termine:**  
18.–29.01.2010  
in jedem ppedv-Schulungsstandort

**Preis:**  
ab 149,- EUR/Pers. zzgl. MwSt.

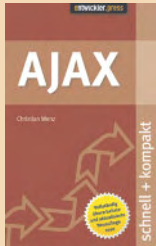
[www.ppedv.de/Quicky](http://www.ppedv.de/Quicky)



480 Seiten  
englisch  
erschienen 11/09  
ISBN:  
978-0-3216-3798-7  
EUR 35,26

**Essential Facebook Development**  
John Maver, Cappy Popp  
Addison-Wesley

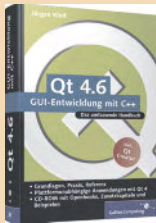
## AJAX



130 Seiten  
deutsch  
erscheint 10/10  
ISBN:  
978-3-86802-045-8  
EUR 12,90

**AJAX**  
überarbeitete u. aktualisierte Aufl. 2010  
Christian Wenz  
entwickler.press

## Qt 4.6



820 Seiten  
deutsch  
erschient 01/10  
ISBN:  
978-3-8362-1542-8  
EUR 49,90

**Qt 4.6**  
GUI-Entwicklung mit C++  
2., aktualisierte und erweiterte Auflage  
Jürgen Wolf  
Galileo Computing

SharePoint-Veranstaltungen  
MicrosoftInformationsveranstaltungen  
zur SharePoint-Entwicklung

In diesen Ganztagesworkshops werden die neuen Möglichkeiten von SharePoint 2010 als Entwicklungsplattform vorgestellt. Es wird diskutiert, wie ein SharePoint-Entwicklungsprozess unter Zuhilfenahme von Microsofts Application Lifecycle Management-Lösungen optimal abgebildet und ausgestaltet werden kann.

25.01.2010  
Bad Homburg

27.01.2010  
Köln

28.01.2010  
München

**Info und Anmeldung:**  
www.event-team.com/  
events/moss2009/default.aspx

»In Memory« ist die Voreinstellung und bewirkt zur Laufzeit, dass das XNA Framework die kompletten Audiodaten in den Speicher lädt, sobald Sie auf die Daten zugreifen. Verwenden Sie diese Option immer für kleine Sounds mit hoch frequentiertem Einsatz. Eine Hintergrundmusik umfasst in der Regel mehrere Minuten, weshalb es nicht sinnvoll ist, die Daten komplett in den Speicher zu laden.

Zum einen benötigt das XNA-Framework eine Weile, um die Daten in den Speicher zu laden, und zum anderen steigt der Speicherverbrauch enorm. Legen Sie stattdessen eine Streaming-Wave-Bank an. Sobald Sie das Musikstück abspielen, liest das XNA-Framework stetig einen Teil der Daten in den Buffer und spielt sie ab. Der Einsatz von Streaming-Wave-Banks impliziert die Wahl einer anderen Konstruktorüberladung der WaveBank-Klasse. Die Signatur dieser Überladung sieht so aus:

```
public WaveBank(AudioEngine audioEngine,  
string streamingWaveBankFilename,  
int offset, short packetSize)
```

Die ersten beiden Parameter sind bekannt. Offset spezifiziert die Position, bei der das XNA-Framework anfängt die Daten in den Speicher zu laden. Die Angabe erfolgt in Sektoren, gemessen an der DVD-Sektorgröße. Mit dem letzten Argument legen Sie die Größe des Buffers fest, den das XNA-Framework stetig mit Daten füllt, um die Audiodaten wiederzugeben. Die Angabe erfolgt ebenfalls in Sektoren. (Sektoren einer DVD umfassen jeweils 2048 Bytes. Der WaveBank-Konstruktor fordert eine Mindestgröße von 2 Sektoren, demzufolge sind 4096 Bytes erforderlich.)

## Sound-Effekte

Mittels der Parameter eines digitalen Signalprozessors (DSP) empfinden Sie die Akustik eines spezifischen Raums nach. Materialien absorbieren und reflektieren Schallwellen in unterschiedlicher Art und Weise. Eine hohe Schallreflektion charakterisiert beispielsweise ein Badezimmer. XACT platziert alle vordefinierten Effekte unterhalb des Knotens »DSP Effect Path Presets« im Projektbaum. Per Drag & Drop weisen Sie einem Sound der Sound-Bank den Effekt zu – ziehen Sie dazu den Sound auf den jeweiligen Effekt im Projektbaum. Zur Konfiguration des Effekts klicken Sie auf dessen Knoten. XACT hält eine Reihe von vordefinierten Effekten bereit. Dazu gehören beispielsweise »Living Room«, »Stone Room«, »Auditorium«, »Concert Hall« oder »Arena«. Seitens der Anwendung sind keine Änderungen notwendig.

## Dreidimensionale Wiedergabe

Bis hier hin lag der Fokus auf der einfachen Wiedergabe von Mono- und Stereo-Audioquellen. Die auditive Komponente unterlag keiner Dynamik, die der Benutzer oder die Spiellogik durch Positions- oder Geschwindigkeitsänderungen bewirkte. Dies ändert sich mit der Positionierung von Audioquellen im dreidimensionalen Raum. XACT regelt in Abhängigkeit von den Parametern automatisch Lautstärke, Balance und Frequenz der Audioquellen. Microsoft hat bei der Implementierung von XACT versucht, Erscheinungen, wie zum Beispiel den Doppler-Effekt, aus der Realität zu simulieren. Der Doppler-Effekt kommt sehr schön zur Geltung, wenn sich ein Polizeiwagen mit Sirene nähert. Als Konsequenz steigen Lautstärke und Tonlage. Nachdem der Wagen an Ihnen vorbeigefahren ist, sinken Lautstärke und Tonlage wieder.

Essenziell sind die Klassen AudioEmitter und AudioListener. Letztere Klasse korrespondiert in der Regel mit den Eigenschaften der Kamera – also mit deren Position, Ausrichtung und Geschwindigkeit. Für jede Audioquelle mit unterschiedlichen Eigenschaften instanzieren Sie ein Objekt vom Typ AudioEmitter. Immer wenn sich die Kameraeigenschaften oder die Eigenschaften der Audioquelle ändern, müssen Sie die Objekte entsprechend anpassen.

```
m_oEmitter = new AudioEmitter();  
m_oListener = new AudioListener();
```

```
m_oListener.Position = Vector3.Zero;  
m_oListener.Forward = Vector3.UnitZ;  
m_oListener.Up = Vector3.Up;  
m_oListener.Velocity = Vector3.Zero;
```

```
m_oEmitter = new AudioEmitter();  
m_oEmitter.Position = new Vector3(100.0f, 0.0f, 0.0f);  
m_oEmitter.Forward = -Vector3.UnitX;  
m_oEmitter.Up = Vector3.Up;  
m_oEmitter.Velocity = Vector3.Zero;
```

Mittels Instanzen der Typen AudioEmitter und AudioListener definieren Sie ausschließlich die Attribute der Audioquellen beziehungsweise des Empfängers. Die Apply3D-Methode eines Cue-Objekts stellt eine Relation zwischen den Attributen und einer spezifischen Audioquelle her. Nachträgliche Attributänderungen haben keinen Einfluss auf die Wiedergabe. Erst nach einem erneuten Aufruf von Apply3D übernimmt die Audio-Engine die Attribute.

```
m_oCue.Apply3D(m_oListener, m_oEmitter);
```

## Variablen

Variablen dienen als Mechanismus, um Sounds in einen vordefinierten Zustand zu bringen. Sounddesigner definieren Variablen mit einem Namen, einem Wertebereich und einem initialen Wert. Variablen können entweder Public oder Private sein. Während Ihre Anwendung auf Basis des XNA Frameworks auf öffentliche Variablen zugreifen und dessen Werte ändern kann, verbirgt XACT private Variablen vor dem Client. Ferner unterscheidet man zwischen impliziten und expliziten Variablen. Implizite Variablen gehören ausschließlich der XACT-Engine selber. Jene Variablen sind zwar änderbar, dürfen aber nicht gelöscht werden. Dazu gehört beispielsweise SpeedOfSound oder DopplerPitchScalar. Beide Variablen beeinflussen die Ausprägung des

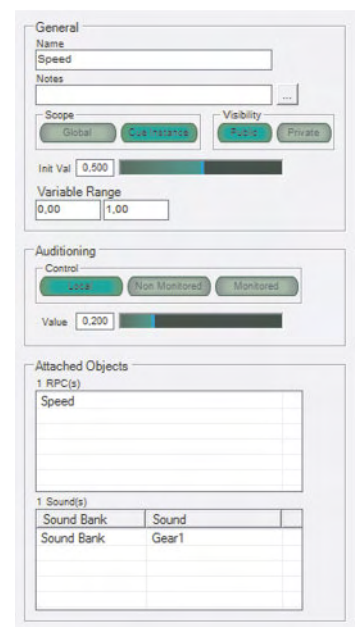


Abb. 4: Eigenschaften einer Variablen

Doppler-Effekts. Variablen, die der Sound-designer im Rahmen eines XACT-Projekts anlegt, gehören dagegen zu den expliziten Variablen.

Variablen erzeugen Sie entweder über das Kontextmenü der Knoten »Variables«, »Global« sowie »Cue Instance« oder mit Hilfe der Befehle »New Global Variable« beziehungsweise »New Cue Instance Variable«. Anschließend öffnet sich in XACT ein Dialog, in dem Sie die Eigenschaften der Variablen definieren. Alternativ können Sie die Variable im Projektbaum auswählen und im Eigenschaftenfenster unten links konfigurieren (siehe Abbildung 4).

Legen Sie nach dem Anlegen einer Variablen deren Gültigkeitsbereich und Sichtbarkeit fest und vergeben Sie einen Wertebereich sowie einen initialen Wert. Während Sie einen Sound im XACT abspielen, können Sie den Wert der Variablen im Eingabefeld »Value« ändern.

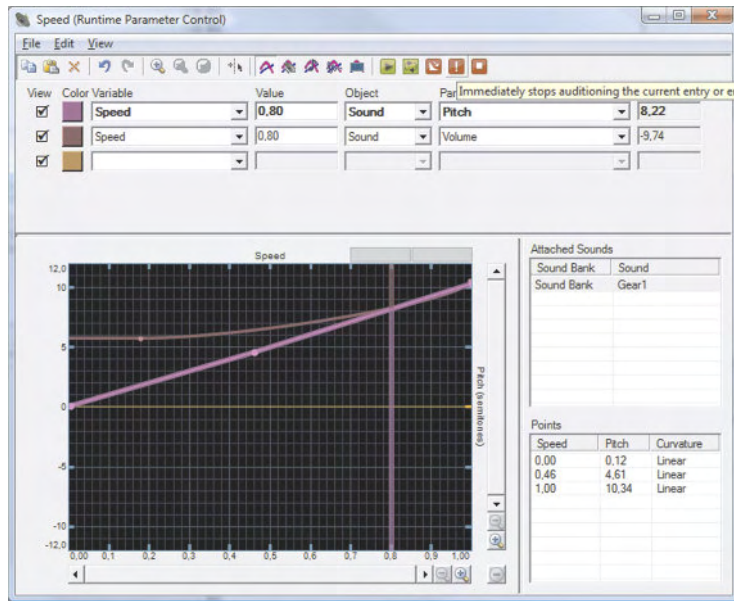


Abb. 5: RPCs definieren die Auswirkung der Variablen

Unten rechts listet der Dialog für alle Punkte der Kurve die Werte der Variablen, die Werte des Parameters und den Kurventyp auf. Alle drei Eigenschaften können Sie manuell ändern. Als Standard werden im XACT lineare Kurven verwendet. Alternativ stehen »Fast«, »Slow« und »Sin/Cos« als Kurventypen zur Verfügung. Der Kurventyp bezieht sich immer auf die Strecke zwischen dem aktuell ausgewählten Punkt und dem Folgepunkt. Zur Laufzeit verwenden Sie die SetVariable-Methode eines Cue-Objekts, um dessen Wert zu ändern. Die Methode erwartet als erstes Argument den Namen der Variablen und als zweites dessen Wert.

```
m_oCue.SetVariable("Speed", m_fSpeed);
```

Globale Variablen sprechen Sie stattdessen über die AudioEngine-Instanz an. Genauer gesagt, Sie verwenden die SetGlobeVariable-Methode mit derselben Signatur.

**Das SoundEffect-API**

Das XNA-Framework offeriert eine Alternative, die auf »herkömmliche« Weise, also im Sinne der Content-Pipeline arbeitet. Konkret bedeutet das, dass Sie die Content-Pipeline mit einer neuen Sounddatei bestücken und zur Laufzeit eine Instanz des Sounds erzeugen. Das XNA-Framework bietet von Haus aus Content-Importer für die Dateiformate Wave, MP3 und WMA. Als Content-Processor fungiert »Sound Effect – XNA Framework«. Zur Laufzeit repräsentiert eine SoundEffect-Instanz die Audiodaten.

```
m_oSoundEffect = this.Content.Load<SoundEffect>("Sounds\\Exp101");
```

Auf den ersten Blick erscheint das API sehr überschaubar. Lediglich zwei Eigenschaften, »Name« und »Duration«, geben Auskunft über den Namen des Assets und der Abspieldauer. Mittels der Play-Methode, von der es zwei Überladungen gibt, spielen Sie die Audiodaten ab. Die erste Überladung ist parameterlos. Die zweite Überladung ermöglicht die Konfiguration der Lautstärke, der Frequenz und der Balance.

```
public bool Play(float volume, float pitch, float pan)
```

Für den Parameter »volume« sind Fließkommawerte im Bereich von 0 bis 1 gültig. Die Angabe der Lautstärke

**MSDN Beginner Developer Learning Center**



Auf MSDN (Microsoft Developer Network) ist ein »Beginner Developer Learning Center« eingerichtet worden. In diesem Portal werden kostenlose, englischsprachige Tutorials für Programmierneinsteiger angeboten. Dort kann man bequem die Entwicklung mit Microsofts .NET-Framework erlernen oder seine Kenntnisse darüber vertiefen.

Das Portal besteht aus vier zentralen Bereichen: »Web Track« für die Webprogrammierung, »Windows Track« zum Entwickeln für Windows-Systeme, »Aspiring Pro« für ambitionierte Projekte und die »Kid's Corner«. Von dem Portal können auch die kostenlosen Visual-Studio-Express-Versionen heruntergeladen werden.

Info: [msdn.microsoft.com/beginner](http://msdn.microsoft.com/beginner)

**Windows Scripting**



500 Seiten  
deutsch  
erschienen 12/09  
ISBN:  
978-3-8273-2910-3  
EUR 29,80

Windows Scripting lernen  
Holger Schwichtenberg  
Addison-Wesley

**Windows Azure**



350 Seiten  
englisch  
erschienen 12/09  
ISBN:  
978-1-4302-2469-3  
USD 39,99

Introducing Windows Azure  
Henry Li  
Apress

**Do-Dotnet 06.01.2010**

.NET 4.0 und VS2010  
Dr. Holger Schwichtenberg

06.01.2010  
18:00 – 20:00 Uhr  
Lünen

Info:  
[www.do-dotnet.de](http://www.do-dotnet.de)



VSone

Die Konferenz zu Visual Studio, .NET, SQL & more

24. - 25. Februar 2010 München

Mehr als 120 Sessions

10 parallel Tracks mit 6 Konferenzen: VSone + 5 weitere Konferenzen = 1 Preis

Große Fachausstellung

Bis 31. Jan. 2010 Early Bird sichern & so mindestens 100,- EUR sparen.

Infos & Anmeldung: www.VSone.de

C++



380 Seiten deutsch erschienen 12/09 ISBN: 978-3-8362-1441-4 EUR 24,90

Spielend C++ lernen André Willms Galileo Computing

dfPUG Hamburg 14.01.2010

C#-Klassenbibliothek StarLib Rudolf Vogel, Mike Michel

14.01.2010 19:00 Uhr

Info:

www.dfpug.de/default.afp?url1=veran/veran.afp?url=regional/hamb/treffen.htm

steht in Relation zur statischen Eigenschaft »MasterVolume« der SoundEffect-Klasse. Die Frequenz manipulieren Sie im Wertebereich von -1 bis 1, wobei 0 der normalen Frequenz entspricht. Die Balance reicht von -1.0 bis 1.0. Ausgeglichen ist die Wiedergabe beim Wert 0.0.

Bei näherer Betrachtung sticht die CreateInstance-Methode ins Auge, die ein SoundEffectInstance-Objekt zurückgibt. Verwenden Sie die diese Methode, bevor Sie die Audiodaten mit Play wiedergeben, um zusätzliche Parameter zu konfigurieren.

```
SoundEffectInstance oInstance = m_oSoundEffect.CreateInstance();
```

SoundEffectInstance-Objekte gewähren die Parameterkonfiguration über entsprechende Eigenschaften. Jede Instanz greift auf die Daten des entsprechenden SoundEffect-Objekts zurück und reichert die Daten um die entsprechenden Parameter-Konfigurationen an. Zum Methodenrepertoire des Typs SoundEffectInstance zählen Apply3D, Pause, Play, Resume und Stop. Analog zur Cue-Klasse gewährt die Apply3D-Methode die Positionierung der Audioquelle.

Während die Zahl der SoundEffect-Objekte lediglich durch den vorhandenen Speicher begrenzt wird, setzen Zune-Geräte den SoundEffectInstance-Objekten ein Limit von 16 gleichzeitig abgespielten Instanzen, bezogen auf alle geladenen SoundEffect-Objekte. Die Xbox 360 bietet insgesamt 300 SoundEffectInstance-Objekte. Auf Windows-Systemen definiert die Speicherkapazität des Systems die Obergrenze. SoundEffectInstance-Objekte werden ungültig, sobald eine SoundEffect-Instanz durch den Garbage-Collector aufgeräumt wird.

Videos abspielen

Eine der neuen Schlüsselfunktionen des XNA Game Studio 3.1 ist die Wiedergabe von Videos. Mit dem XNA-Framework 3.1 wird Ihnen sowohl die einfache, direkte Wiedergabe von Videos geboten, als auch deren Projektion auf Texturen, so dass Sie sie in eine dreidimensionale Welt einbetten können. Die Content-Pipeline stellt eine Hand voll Anforderungen, denen importierte Videos genügen müssen:

- Digital Rights Management (DRM) darf nicht aktiv sein.
■ Das Video muss als WMV-9, dem VC-1 Standard, kodiert sein.
■ Das Video muss eine konstante Bitrate (CBR) besitzen.
■ Das Video muss über eine Audiospur verfügen.

- Die Audiospur muss im Windows Media Audio-Format (WMA) kodiert sein und eine konstante Bitrate haben.
■ Maximale Auflösung von 1280x720 Pixel (720p) bei 30Hz.

Genügt das Video nicht diesen Anforderungen, schlägt der Kompilervorgang fehl. Zur Laufzeit repräsentiert die Klasse »Video« die WMV-Datei. Laden Sie das Asset wie gewohnt innerhalb der LoadContent-Methode. Ferner instanzieren Sie die VideoPlayer-Klasse, deren Konstruktor parameterlos ist. Ein VideoPlayer-Objekt ist für die Wiedergabe verantwortlich.

```
protected override void LoadContent()
{
    m_oVideoPlayer = new VideoPlayer();
    m_oVideo = this.Content.Load<Video>("Burnout");
}
```

VideoPlayer stellt die üblichen Methoden zum Abspielen, Pausieren, Fortsetzen und Anhalten des Videos zur Verfügung. Ferner lassen sich der Ton (de)aktivieren und das Video in einer Schleife abspielen.

```
if (m_oVideoPlayer.State == MediaState.Stopped)
{
    m_oVideoPlayer.IsLooped = true;
    m_oVideoPlayer.Play(m_oVideo);
}
```

Vorausgesetzt, die VideoPlayer-Klasse spielt ein Video ab, das heißt, der Wert der Eigenschaft »State« ist MediaState.Playing, dann liefert GetTexture den aktuellen Frame als Texture2D-Instanz. Andernfalls löst die Klasse eine InvalidOperationException aus. Entweder Sie rendern den Frame als Sprite oder Sie verwenden den Frame als Textur eines dreidimensionalen Körpers.

```
protected override void Draw(GameTime gameTime)
{
    GraphicsDevice.Clear(Color.CornflowerBlue);

    if (m_oVideoPlayer.State == MediaState.Playing)
    {
        Texture2D oTexture = m_oVideoPlayer.GetTexture();

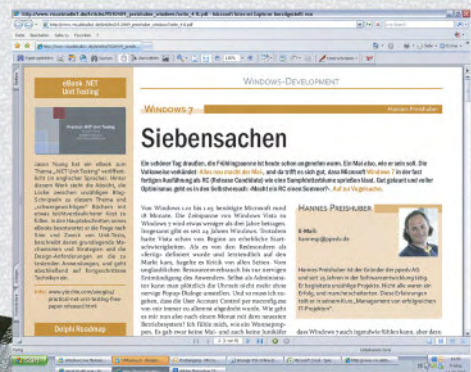
        spriteBatch.Begin();
        spriteBatch.Draw(oTexture, new Rectangle(0, 0, this.GraphicsDevice.Viewport.Width, this.GraphicsDevice.Viewport.Height), Color.White);
        spriteBatch.End();

        base.Draw(gameTime);
    }
}
```

Ihnen stehen damit alle Möglichkeiten offen: Von der einfachen Platzierung des Videos in der dreidimensionalen Welt bis hin zur Nachbearbeitung durch Post-Processing-Effekte.

Die Online-Datenbank auf www.VisualStudio1.de

\*Jetzt PDFs mit klickbaren Links!



Nur für Abonnenten: alle Artikel bisheriger Ausgaben und Downloads.