

Verarbeitung von Audio und Video in .NET

Herfried K. Wagner
Microsoft MVP für Visual Basic



APIs zur Medienverarbeitung in .NET

- Programmierschnittstellen von Microsoft
 - DirectX (DirectX 9.0)
 - DirectShow
 - DirectSound
 - DirectMusic
 - AudioVideoPlayback
 - ...
 - Windows Media (Windows Media 9 Series)
 - Windows Media Format
 - Windows Media Encoder
 - Windows Media Player
 - ...

Zugriff auf Schnittstellen in .NET

- Verwaltete Assemblies (Ideallösung)
- Bestehende Bibliotheken (Kompromiss)
 - COM Interop
 - Primary Interop Assemblies (PIAs)
 - Platform Invoke (PInvoke)
- Managed Extensions for C++
 - Erweiterung der Programmiersprache C++ für die Common Language Runtime (CLR)
 - Nahtloses Mischen von verwaltetem und unverwaltetem Code mit It Just Works (IJW)

Medienverarbeitung in .NET

- .NET ist Technologie für User-Mode
 - Keine direkten Hardwarezugriffe
 - Kein garantiertes Zeitverhalten
 - → Ist Erweiterung/Nutzung der APIs in verwaltetem Code sinnvoll?
- Verfügbarmachen bestehender APIs für .NET
 - Verwaltete Wrapper um bestehende COM-Komponenten
 - (Zeitkritische) Implementierungen unverändert
 - Ggf. Kombination mit Managed Extensions for C++

DirectX

- Sammlung von Programmierschnittstellen zur Entwicklung von Multimediaanwendungen unter Microsoft Windows
 - Medienverarbeitung: DirectShow, DirectSound, DirectMusic, AudioVideoPlayback
 - Weiters: DirectX Graphics, DirectInput, DirectPlay...
- Programmierschnittstellen
 - Unverwaltetes DirectX
 - Verwaltetes DirectX („Managed DirectX“)
 - Nur Teilbereiche des unverwalteten DirectX abgedeckt

DirectShow

- Architektur für Streaming von Audio und Video
- Wiedergabe, Aufnahme, Formatkonvertierung
- COM-basierend
- Nutzung:
 - COM Interop mit Visual Basic-Bibliothek (nur Abspielen)
 - Nachbilden der COM-Schnittstellen in einer verwalteten Programmiersprache (aufwendig)
 - Managed Extensions for C++ (unkomfortabel)

DirectSound

- Wiedergabe von Audio mit geringer Latenz und Aufnahme
- Hohes Maß an Kontrolle über verwendete Hardwareressourcen
- Mehrere Klänge gleichzeitig wiedergeben
- Anwenden von Effekten auf Audiodaten
 - Vordefinierte Effekte
 - Eigene Effekte als DirectX Media Objects (DMOs)
- Nutzung: PIA

Beispiel: DirectSound 8 vs. DirectSound 9

- API an .NET-Konventionen angepasst

```
Dim dx As New DirectX8()  
Dim ds As DirectSound8 = dx.DirectSoundCreate("")  
Dim bd As DSBUFFERDESC  
ds.SetCooperativeLevel (Me.Handle.ToInt32(),  
CONST_DSSCLFLAGS.DSSCL_NORMAL)  
bd.Flags = CONST_DSBCAPSFLAGS.DSBCAPS_CTRLVOLUME  
Dim b As DirectSoundSecondaryBuffer8 = _  
    ds.CreateSoundBufferFromFile("C:\WINDOWS\Media\tada.wav", bd)  
b.SetVolume(-100)  
b.Play(CONST_DSBPLAYFLAGS.DSBPLAY_DEFAULT)
```



```
Dim dev As New Device()  
dev.SetCooperativeLevel (Me, CooperativeLevel.Normal)  
Dim bd As New BufferDescription()  
bd.ControlVolume = True  
Dim b As New SecondaryBuffer("C:\WINDOWS\Media\tada.wav", bd, dev)  
b.Volume = -100  
b.Play(0, BufferPlayFlags.Default)
```

DirectMusic

- Bearbeitung und Wiedergabe von MIDI und Downloadable Sounds (DLS)
- Ereignisse bei Wiedergabe über DirectX-Ereignisse
- Nutzung:
 - In DirectX 9.0 beschränkt auf C/C++
 - Evtl. Lösung mit Managed Extensions for C++
 - COM Interop auf Visual Basic-Bibliothek (Probleme bei Praxistests)

AudioVideoPlayback

- Wiedergabe von Audio und Video
 - Klassen Audio und Video
- Ereignisgesteuerte Darstellung des Videos über Direct3D-Texturen
- Ereignisse zur Statusüberwachung bei Wiedergabe
- Nutzung: Verwaltete Assemblies (problematisch)

Windows Media

- Operationen
 - Aufnahme
 - Wiedergabe
 - Bearbeitung und Codierung
- Formate
 - Windows Media
 - Formate anderer Hersteller
- Arbeitet zusammen mit DirectShow
 - DirectX Media Objects (DMOs)

Windows Media Format

- Verarbeitung von Daten im Advanced Systems Format (ASF)
 - Containerformat für streamingfähige Medien
- Codierung und Verarbeitung von Metadaten
- Nutzung: Managed Extensions for C++

Windows Media Encoder

- Codieren von Audio und Video
 - Flexibel mit verschiedenen Codecs benutzbar
- Profile
 - Medientyp des codierten Ausgabestroms, Bitraten etc. des Zielpublikums
- Kontrolle über Codierungsvorgang
 - Ereignisse für Fortschritt, Status und Fehler
 - Eigene Benutzerschnittstelle
- Nutzung: COM Interop

Windows Media Player

- ActiveX-Steuererelement des Windows Media Player
- Medienverwaltung
 - Medienbibliothek
 - Wiedergabelisten
 - Metadaten
- Wiedergabe von Audio und Video
 - Kontrolle über Aussehen des Players
 - Eigene Benutzerschnittstelle möglich
- Nutzung: COM Interop, zuvor PIA

Zusammenfassung und Ausblick

- **Verwaltete Unterstützung**
 - Nur für einfaches Abspielen und Aufnahme
 - Unvollständige Wrapper → keine vollständig verwalteten Lösungen möglich
 - Komplexere Lösungen nur mit Managed Extensions for C++
- **Zukünftige Entwicklungen**
 - Abdecken von User-Mode-APIs durch Wrapper bzw. PIAs
 - Low-Level-APIs weiterhin auf COM/C++ beschränkt?
 - Neue APIs (wie AudioVideoPlayback)?

**Vielen Dank für Ihre
Aufmerksamkeit!**

