Streaming Media im Internet

Als Streaming Media wird eine Technologie bezeichnet, die es erlaubt, multimediale Objekte wie Live-Präsentationen, Videofilme oder Audiosequenzen auf ressourcenschonende Weise zu präsentieren und dabei dem Nutzer die Möglichkeit gibt, durch Vor- und Zurückspulen in den Abspielvorgang einzugreifen. Der Artikel klärt grundlegende Begriffe wie HTTP- und RTSP-Streaming, Multicast und Unicast, gibt Auskunft über Produkte und Hersteller der verschiedenen Softwarekomponenten und zeigt, wie derartige Objekte in WWW-Seiten eingebettet werden können. Den Abschluss bilden Szenarien zu verschiedenen Anwendungsmöglichkeiten und Hinweise zu den an der Universität nutzbaren Streamingservern.

Das Internet ist gut geeignet, multimediale Daten für seine Teilnehmer bereitzustellen. Es kommt aber ebenso darauf an, akzeptable Antwortzeiten und genussvolle Präsentationen zu erhalten. Hier ist einerseits eine möglichst breite Auffahrt zur Datenautobahn gefragt und andererseits ein ebenso breiter Weg bis hin zu den Daten. Selbst wenn alle Bedingungen bestens erfüllt sind, macht es wenig Sinn, einen Spielfilm mit 90 Minuten Spieldauer zunächst herunterzuladen und Plattenplatz damit zu belegen, um anschließend – nach eventuell nervenaufreibender Download-Zeit – festzustellen, dass der Inhalt gar nicht so spannend ist und man das Geld für die benötigte Ladezeit besser gespart hätte. Gesucht ist eine Technologie, die sofort mit dem Abspielvorgang beginnt und jederzeit analog zum Videorecorder Vor- und Rückspulmöglichkeiten bietet. Als ein Weg erscheint hier die mit „Streaming Media“ bezeichnete Technologie, die es ermöglicht, im Internet abgelegte Multimediainhalte (Audio, Video, Text, Animationen ...) in Echtzeit verfügbar zu machen, ohne Wartezeiten für ein „download“ des gesamten Materials zu benötigen.

Vorteile von Streaming Media

Streaming-Media bietet dem Betrachter einer Web-Seite entscheidende Vorteile: Die Multimediadaten müssen nicht auf die lokale Festplatte kopiert werden. Beim Streaming werden die Inhalte, z. B. Video und Audio, paketweise übertragen und beim Empfänger (Client-Software) in einen Puffer gespeist, der vor Beginn der Präsentation zunächst mit Daten gefüllt wird.Das führt zu einer kurzen Wartezeit, aber der Puffer dient zur Sicherung eines kontinuierlichen Abspielvorgangs. Anschließend wird mit dem Abspielen des Medienstreams begonnen. Während der Übertragung weiterer Pakete wird am lokalen Rechner der Puffer bereits ausgelesen und in diesem Beispiel als synchrones Video und Audio dargestellt. Der Spielfilm wird also schon angezeigt, während weitere Datenpakete in den Puffer nachgeladen werden. Die Daten werden durch die Client-Software in dem Takt aus dem Puffer entnommen, in welchem sie benötigt werden. Unser Puffer versagt seinen Zweck, wenn die tatsächlich nachgeladende Datenmenge stets geringer ist, als die zum Abspielvorgang benötigte. Dann schrumpft der Pufferinhalt bis zu dem Punkt, an dem die Software jedes Paket sofort verwenden muss: Die Bilder ruckeln, der Ton stockt, der Puffer muss erneut gefüllt und anschließend präsentiert werden.

Komponenten dieser Technologie

Die Streaming-Technologie verlangt im wesentlichen drei Komponenten: die Encoder, die Server und die Clients. Ausgehend von den digitalen Videos stellt ein Encoder eine Software dar, die die Daten passend komprimiert. Authoring-Werkzeuge vereinen Video- und Audio-Spuren, Untertitel, anklickbare Internet-URLs und ähnliches. Ein Server schickt die Ergebnisse ins Netz. Beim Empfänger muss dann eine Client-Software den Kontakt zum Server aufbauen, dessen Daten empfangen, dekomprimieren und abspielen.

Streaming-Varianten

Im einfachsten Fall, dem HTTP-Streaming, können die Videos zusammen mit den dazugehörigen HTML-Seiten von einem herkömmlichen Webviewer verbreitet werden. Zur Übertragung dient dann das HTTP (Hypertext Transfer Protocol). Diese Lösung krankt an einigen Beschränkungen: Ein Webserver ist eigentlich zum Herunterladen kompakter Inhalte gedacht, insbesondere HTML-Seiten sowie GIF- und JPEG-Bildern. Dagegen verlangen Audio und Video andere Fähigkeiten. Während ein Webserver einen Mediastream von Anfang bis Ende ohne Lücken und ohne Wiederholungen abspielt, möchte man bei Audio- und Videostreams vor- und zurückspulen können. Schlimmstenfalls möchte man z. B. von einem zweistündigen Werk nur die im Abspann erwähnten Produzenten sehen.

Außerdem sorgt HTTP für eine sichere Übertragung mit dem zugrunde liegenden TCP. Der Client quittiert jedes Datenpaket; kommen Pakete nicht an, schickt der Server sie erneut. Die Folge davon ist ein großer Aufwand mit gelegentlichem Netzstau.

Zusätzlich zum Webserver verwendet man deshalb einen für Audio und Video spezialisierten Server („Streamingserver“). Das kann ein besonders ausgestatteter Spezialrechner sein, aber ebenso eine Softwarelösung. Als übliches Internet-Protokoll für Streaming etabliert sich RTP (Real-Time Protocol). Es

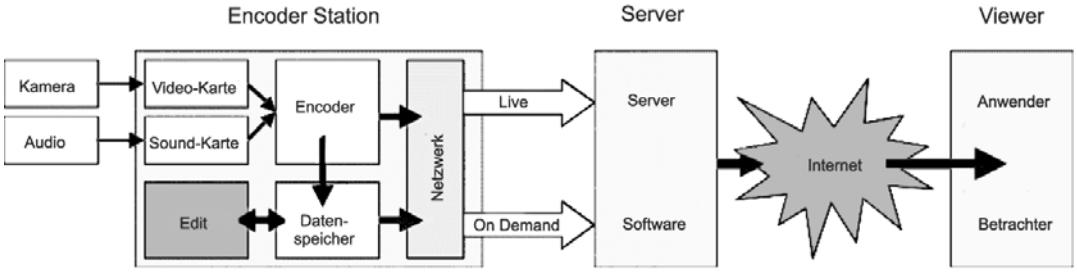

Abb. 1: Live Streaming Prinzipskizze

basiert auf dem einfacheren UDP (User Datagram Protocol), das auf die Empfangsbestätigungsmeldung verzichtet. Die Datenpakete besitzen einen Zeitstempel, so dass sie zeitsynchron dargestellt werden können. Fallen Pakete aus, kann die Abspielsoftware geeignet über die fehlenden Pakete hinweggehen. UDP entlastet das Netz auf diese Weise und man kann es auch schonend bei Live-Übertragungen per Multicast nutzen. Neben RTP zum Transport der reinen Daten wird noch das RTSP (Real-Time Streaming Protocol) zur Positionierung innerhalb der abgelegten Multimediadaten eingesetzt, man redet in diesem Zusammenhang auch von „RTSP-Streaming“. Eine besondere Variante des RTSP-Streamings ist das Livestreaming. Hierbei werden keine auf den Festplatten des Servers abgelegten Daten zum Client geschickt, sondern die Daten werden von einen Encoder live zum Server übertragen, dort kurz zwischengespeichert und an die anfragenden Clients weitergesandt. Der Encoder liefert kontinuierlich die neuesten Daten nach (siehe Abb. 1).

Steigt die Streamnachfrage weiter, bieten Serverprodukte unterschiedliche Erweiterungen. Das Clustering fasst mehrere Videoserver zu einem Verbund zusammen, um die Netzlast auf mehrere Server zu verteilen und sorgt für die automatische Replikation eines neu eingespielten Videos auf alle verfügbaren Server. Clustering ist bei On-Demand-Systemen sinnvoll, wenn es darum geht, auf einen Pool häufig genutzter Videos zuzugreifen. Bei Live-Übertragungen ist das Splitting geeigneter. Dieser Mechanismus erzeugt eine Kette von hintereinander geschalteten Servern, die einerseits das zu übertragende Live-Material an den nächsten Server weiterreichen und andererseits gleichzeitig anfragende Clients mit dem Material versorgen. Uni- bzw. Multicast kennzeichnet die Verbindung zwischen Client und Server. Unicast kennzeichnet eine Punkt-zu-Punkt-Verbindung mit dem Vorteil, dass der Client dem Server Anweisungen zukommen lassen kann, z.B. eine Pause zu machen oder zurückzuspulen und eine Wiederholung des Inhalts vorzunehmen. In On-Demand-Szenarien ist das gewünscht. Der Nachteil besteht in sehr vielen Verbindungen und mehrfachem Senden gleicher Daten. Der Multicastbetrieb, bei dem hingegen nur eine einzige Kopie des Streams über das Netz geschickt wird, ist effektiver, hat jedoch den Nachteil, dass der Client keinen Einfluss auf die zu empfangenen Daten hat. [1] Für Live-Streams ist diese Methode besonders gut geeignet.

Die Anbieter dieser Technologie

Die am weitesten verbreiteten Lösungen für Streaming-Software stammen von RealNetworks [3], Apple [4] und Microsoft [5]. Alle drei bieten Lösungen für Encoder-, Server- und Clientsoftware in unterschiedlichsten Versionen an.

Die Encoder

RealNetworks’ kostenloses Kompressions- und Autorenwerkzeug RealProducer Basic 7.0 verarbeitet wahlweise Dateien oder live eingespeiste Signale. Obendrein erzeugt es gleich fertige Webseiten. Die Basic Variante des Encoders ist kostenlos, während die Pro-Variante für $ 150 zu haben ist.

Um *Apples* QuickTime Movies (.mov) über einen Streamingserver abspielen zu können, müssen diese Movies vorher entsprechend aufbereitet werden. Das Stichwort hierzu heißt „Movie to Hinted Movie“ [7]. Der QuickTime Player Pro besitzt die entsprechenden Export-Funktionen für einen Webserver (‘fast start movie’) oder für einen Streamingserver (‘hinted movie’). Im zweiten Fall legt QuickTime im Film ab, wie die Multimedia-Ströme auf Internet-Datenpakete verteilt werden sollen. Auf Apples Webseite [6] findet man das kostenlose MakeRefMovie, das eine QuickTime-Referenz-Datei erzeugt, die je nach im Player gewählter Bandbreite zu Videos verschiedener Bandbreite verzweigt. QuickTime verfügt über diverse Kompressoren/Dekompressoren (Codecs) für Video und Audio. Keyframebasierte und zugleich hochwertige Kompression bewerkstelligt davon nur der Sörensen-Video-Codec. Er wird typischerweise mit dem QDesign-Music-Codec kombiniert, der MP3-ähnliche Ergebnisse liefert. Die optional erhältliche Developer Edition 2.1 des Sörensen-Codec komprimiert schneller und setzt Keyframes automatisch [2]. Als geeignetes Werk-

zeug zur Datenformatumwandlung kommt ebenfalls die Software Media Cleaner Pro in Frage. Zur Lifestreamübertragung bietet sich das Programm Sörensen Broadcaster an.

Bei *Microsoft* gibt es kostenlos Kompressions- und Autorenwerkzeuge unter dem Namen Windows Media Tools. Es gibt darin Kommandozeilenprogramme, die .avi- und .wav-Dateien ohne Neukompression in das Streamingformat (.asf advanced streaming format) verpacken; ein Plugin exportiert .asf-Dateien direkt aus dem Videoschnittprogramm Adobe Premiere.

Der Windows Media Encoder aus dem Paket erzeugt .asf-Filme, welche optional die Videos für mehrere Bandbreiten parallel enthalten. Wahlweise komprimiert diese Software live eingespeistes Material und sendet es direkt an Microsofts Streamingserver. Es sind verschiedene Codecs für Audio und Video wählbar. Zum Vor- und Zurückspulen benötigt Windows Media einen Index in der .asf-Datei, der sich im Media Encoder automatisch erzeugen lässt.

Die Server

Marktführer RealNetworks, am längsten auf dem Markt, gibt eine kleine Version seines Servers RealServer Basic 7.0 gratis ab, sonst fallen für die Server nicht unerhebliche Kosten entsprechend der Anzahl der abrufbaren Streams an. Die Basic-Variante unterstützt 25 Streams, d. h. es können maximal 25 gleichzeitig aktive Clients versorgt werden. Für 60 Streams (RealServer Plus 7.0) sind schon $ 2000 erforderlich und für die maximal ausgestattete Server-Software, die unter Windows NT, Linux und Solaris läuft, werden $ 80.000 fällig.

Apple bietet kostenlose Streamingserversoftware für Mac OS X und für verschiedene Unixplattformen an. Der QuickTime Streamingserver (Darwin Streamingserver) arbeitet modular als RTSP & RTP Server (zwei Serversubsysteme). Es gibt inzwischen Versionen für verschiedene Unix-Varianten und Windows NT von der Firma Entera (TeraCAST Streamingserver http://streaming.entera.com/latestversion.shtml).

Microsoft liefert die Serversoftware für Windows NT und Windows 2000 ebenfalls kostenlos.

Eine Basic-Server-Variante von RealNetworks und ein QuickTime Streamingserver [9] laufen im Rechenzentrum jeweils unter Solaris 2.7. Ein Server von Microsoft wird gegenwärtig nicht betrieben.

Die Clients

Die drei Hauptanbieter geben in der Grundversion auch kostenlos ihre jeweilige Clientsoftware für verschiedene Betriebssysteme ab. Als eigenständige Programme dienen alle drei Player dazu, Dateien von lokalen Laufwerken oder von Internet-URLs zu öffnen. Als „Kanäle“ , oder „Favoriten“ merken sie häufig genutzte URLs vor. Alle drei Player integrieren sich in die Web-Browser Internet Explorer und Netscape Navigator, siehe auch weiter unten. Kompatibel sind die Systeme von Real, Apple und Microsoft nicht. Jedes der drei Systeme setzt für den gesamten Streamingprozess die eigenen Softwarelösungen voraus. Bei *RealNetworks* gibt es den Real Player 7 Basic (für Windows, Mac OS, Solaris 2.7 und Linux). *Apples* QuickTime Player 4.1 ist in der Grundversion in Mac OS integriert und für Windows erhältlich. Die etwas besseren Versionen zum Mitschnitt und mit weiteren Einstellmöglichkeiten bieten RealNetworks und Apple jeweils zum Preis von $ 30. *Microsoft* liefert den Windows Media Player 7 Beta für Windows und für Mac OS.

Um einen optimalen Datenfluss zu sichern, sollte der übertragene Strom der durchschnittlichen Bandbreite der Verbindung entsprechen. Auf vielen Webseiten findet man deshalb oft verschiedene Video-Versionen etwa für 28,8-kBit/s-Modems oder Hochgeschwindigkeitszugänge. Eigentlich sind aber alle drei großen Lösungen in der Lage, verschieden aufwendige Versionen des Videos aus derselben Steuerdatei anzusprechen. Für pixelgenaues Webseiten-Layout ist diese automatische Auswahl allerdings unglücklich, denn man will die abgespeckten Varianten eines Videos im Allgemeinen mit kleineren Abmessungen darstellen als die Breitband-Version. Welche Version benutzt wird, richtet sich bei Real und bei Apple nach der Übertragungsrate, die der Anwender in der Client-Software eingestellt hat (RealPlayer: Optionen/Einstellungen/Verbindung, QuickTime: Seite „Connection Speed“ im QuickTime-Kontrollfeld); Windows Media bestimmt die Verbindungsgeschwindigkeit selbst. Real und Windows Media wählen nicht nur zu Beginn der Übertragung eine passende Version des Videos, sondern schalten auch nachträglich um, zum Beispiel, wenn die Netzperformance nachlässt. Bei Real betrifft dieses „SureStream“ Audio und Video gleichermaßen. Das Microsoft-System „Intelligent Streaming“ schaltet dagegen nur zwischen verschiedenen Video-Strömen um, fallen Audio-Daten aus, versucht der Player, die Fehler schönzurechnen. Die dynamische Anpassung der Bandbreite gelingt beiden Systemen nur, wenn ein Streamingserver die Videos liefert und kein Webserver.

Streaming Media im WWW

Multimedia-Dateien können stets mit einer zugehörigen, sogenannten Helper-Applikation betrachtet werden, Real-Movies mit dem Real Player, QuickTime-Movies mit dem QuickTime-Player, Windows Media Files mit dem Windows Media Player, usw. Dabei ist immer die Kenntnis der genauen Adresse des zu betrachtenden Files erforderlich. Einfacher für den Benutzer gestaltet es sich, wenn er auf einer Web-Seite lediglich einen vorbereiteten Link anklicken muss,

ohne zu wissen, welchen Standort und welches Format die dahinterliegende Datei hat.

Der folgende Abschnitt soll zeigen, wie Multimedia-Files nahtlos in HTML-Seiten eingebunden werden können.

Das <object>-Tag

<object>

Zum Einbetten von Multimedia-Objekten in eine HTML-Datei benutzt man das <object>-Tag.

Beispiel:

```
<object data="beispiel.mov"></object>
```

So wird es vom World Wide Web Consortium (W3C) in der Spezifikation des HTML-Standards in der Version 4.01 vom 24.12.1999 festgelegt.[13]

Dieses Tag wird derzeit jedoch noch nicht von allen Browsern (z. B. Internet Explorer 4) unterstützt, man sollte deshalb zusätzlich die herkömmliche Art der Einbettung von Objekten mit dem <embed>-Tag (ursprünglich eine Erfindung von Netscape) anbieten.

Beispiel:

```
<object data="beispiel.mov"
        type="video/quicktime">
    <embed src="beispiel.mov"> </embed>
</object>
```

Für ältere Browser, die auch <embed> nicht verstehen, kann mit Hilfe des <noembed>-Tags ein Alternativinhalt angegeben werden.

Beispiel:

```
<object data="beispiel.mov"
        type="video/quicktime">
    <embed src="beispiel.mov"> </embed>
    <noembed>
      <img src="Buch.gif">
    </noembed>
</object>
```

Attribute des <object>-Tags

In das <object>-Tag können verschiedene Attribute aufgenommen werden, die die Anzeige des Multimedia-Files steuern. Festgelegt werden können z. B. Ausmaße des Anzeigefensters, Bildschirmausrichtung, Hinweise zum benötigten Plugin und Angaben dazu, ob die Anzeige manuell oder automatisch (autostart) bzw. einfach oder wiederholt (loop) erfolgen soll. Je nach Anwendung können weitere Parameter hinzukommen.

MIME-Type

(**M**ultipurpose **I**nternet **M**ail **E**xtensions für Auswahl der entsprechenden „Helper Application“)

type = Mime-Type, z. B. video/quicktime

Größen, Rahmen und Abstände

height = Höhe in Pixel oder Prozent
width = Breite in Pixel oder Prozent
border = Rahmenbreite in Pixeln
vspace = vertikaler Abstand zum Objekt in Pixeln oder Prozent
hspace = horizontaler Abstand zum Objekt in Pixeln oder Prozent

Ausrichtung

align = left | right | middle | top | bottom

Plugin-Vermerke

pluginurl = URL, die die Installation des erforderlichen Plugins übernimmt (immer Endung .jar)
pluginspage = URL, auf der Installationshinweise für das Plugin bereitstellt sind

Sonstige

hidden = „true“ | „false“
autostart = „true“ | „false“
loop = „true“ | „false“
palette = „foreground“ | „background“
standby = „Plugin wird geladen ...“

Beispiel:

```
<object
  data="http://www.hu-berlin.de/video/
       IB11.mov"
  type="video/quicktime"
  pluginspage="http://www.apple.com/
               download/"
  autostart="false"
  loop="true"
  standby="Bitte warten..."
  border="2"
  width="322"
  height="260">
  <embed
  src="http://www.hu-berlin.de/video/
      IB11.mov">
  </embed>
    <noembed>Hier sollte eigentlich ein
    Movie erscheinen.
  </noembed>
</object>
```

HTTP-Streaming

Das oben angeführte Beispiel setzt voraus, dass sich HTML-Seite und Objektdatei, also z. B. das Video, auf dem WWW-Server befinden. In die Serverkonfiguration müssen folgende MIME-Types eingetragern werden:

MIME-Type	Dateiendungen
audio/x-pn-realaudio	ra ram
audio/x-pn-realaudio-plugin	rpm
video/quicktime	qt mov
video/x-ms-asf	asf asx

(Diese Eintragungen müssen sich auch am darstellenden Client wiederfinden. In der Regel übernehmen die Installationstools der Plugins diese Aufgabe.) Beim Aufruf der HTML-Seiten übernimmt der WWW-Server den Transport der Daten über das HTTP-Proto-

koll. Dabei wird zunächst das Video geladen und anschließend dargestellt. Für den WWW-Server ist die Aufbewahrung und Darstellung derartiger Files eine große Belastung. Aus diesem Grund sollten Files, die häufig angezeigt werden, auf einen separaten Streamingserver ausgelagert werden.

RTSP-Streaming

Das Einbetten von Multimedia-Files, die auf einem Streamingserver abgelegt wurden, ist etwas komplizierter. Eine Adresse zu einem solchen File kann beispielsweise so aussehen: *rtsp://helios.rz.hu-berlin.de :5540/haerdle23.rm*

Diese Adresse kann jedoch nicht direkt in einen Link auf einer HTML-Seite eingefügt werden, da dort das RTSP-Protokoll nicht interpretiert werden kann. In den Link kann also nur eine Datei aufgenommen werden, die über das HTTP-Protokoll aufgerufen werden kann. Diese Zwischendatei enthält dann einen Link auf die Datei des Streamingservers.

Beispiel 1: Einbetten eines Real-Movies in eine Web-Seite

```
<object data="haerdle.rpm"
  width="196" height="162"
  autostart="true"
  controls="ImageWindow"
  console="_master">
</object>
<br>
<object data="haerdle.rpm"
  width="44" height="26"
  controls="PlayButton" console="video1">
</object>
<object data="haerdle.rpm"
  width="26" height="26"
  controls="StopButton" console="video1">
</object>
```

Bemerkung: Zur Steuerung des Movies können alle Control-Elemente einzeln angegeben werden.

Beispiel 2: Einbetten eines QuickTime-Movies in eine Web-Seite

```
<object data="zwischendatei.mov"
  type="video/quicktime"
  pluginspage="http://www.apple.com/
  download/"
  autostart="true"
  loop="false"
  width="192"
  height="162">
</object>
```

Die Datei zwischendatei.mov sieht wie folgt aus:

```
rtsptext
rtsp://helios.rz.hu-
                  berlin.de/IB/IB22.mov
```

Bemerkung: Wenn die Steuerungsleiste angezeigt werden soll, müssen zur Höhe (height) des Movies 18 Pixel addiert werden. Die Endung der Zwischendatei muss .mov sein! Es handelt sich dennoch um eine normale Textdatei.

Beispiel 3: Einbetten eines Windows Media Files in eine Web-Seite

```
<object data="beispiel.asx"
  type="video/x-ms-asf"
  pluginspage="http://www.microsoft.com/
    windows/windowsmedia/de/download/"
  ... >
</object>
```

Die Datei zwischendatei.asx sieht wie folgt aus:

```
<asx version="3">
    <entry>
        <ref href="beispiel.asf">
    </entry>
</asx>
```

SMIL

Ebenfalls vom W3-Consortium verabschiedet wurde ein Standard für eine **S**ynchronized **M**ultimedia **I**ntegration **L**anguage (SMIL). SMIL ist eine als XML-Applikation entwickelte eigene Sprache, die wie HTML leicht erlernbar und editierbar ist. Sie ermöglicht die Kombination und Synchronisation verschiedener Multimedia-Elemente wie Audio, Video, Text und Grafik. SMIL-Dateien können mit Java applets/servlets oder CGI-Skripten, die auf eine Datenbank zugreifen, verknüpft werden.[14] Zugleich erlaubt SMIL eine bandbreitenfreundliche Verknüpfung von Multimedia-Elementen, indem redundante Informationen nur einmal geladen und dann mehrfach dargestellt werden können.[15]

Wer SMIL erlernen möchte, findet auf den Seiten des W3C diverse Tutorials.
http://www.w3.org/AudioVideo/#Getting

Anwendungsszenarien

Die großen Anbieter von Audio und Video im Web haben ihre eigenen Datenautobahnen bis hin zum Anwender: Server rund um die Welt, welche die Daten spiegeln. Ohne dass der Anwender es merkt, wird er nicht von der eingetippten Adresse etwa aus den USA bedient, sondern von einem näher liegenden Rechner. Idealerweise steht mindestens ein solcher Server am Backbone jedes größeren Internet-Providers. Shooting-Star dieser Art des Audio- und Video-Cachings ist die Firma Akamai (www.akamai.com). RealNetworks, Pionier der Audio- und Video-Übertragung über das Internet, bietet mit dem Real Broadcast Network (www.rbn.com) einen ähnlichen Service für die Verbreitung von Audio und Video an. In Europa will RealNetworks mit Madge.web zusammenarbeiten

(www.madge.com). Eigene Cache-Server betreibt die IntelliVision INETV AG, die zur CEBIT 2000 den Computerkanal comm-TV (www.comm-tv.com) startete. In Deutschland wird das Programm an acht Punkten ins Internet gespeist; für die USA nutzt man die 650 Server der Yahoo-Tochter broadcast.com. [2] Solange Internet-Zugänge per V.90-Modem und ISDN noch die Norm sind, stellt Internet-Video keine Konkurrenz zu Fernsehen, Videokassette oder gar DVD dar. Stattdessen dient es vielen Anbietern dazu, Nischen auszufüllen, die das herkömmliche TV selbst mit Spartenkanälen nicht erreicht.

Auch für uns gibt es interessante Nutzungsmöglichkeiten:

- Live-Übertragung von Veranstaltungen
- Gestaltung von WWW-basierten Lehr- und Lernangeboten
- Aufzeichnung von Vorträgen bzw. aktuellen Veranstaltungen und deren Angebot im Web

Zukünftige Streamingtechnologien werden auch interaktive 3D-Web-Lösungen einbinden. Erste Ansätze hierzu gibt es bereits, beispielsweise von der Firma Shout Interactive[10], oder 3D-Modellierung und interactive Streaming von Pulse Entertainment [11] sowie die 3D- Päsentationsmöglichkeiten im DFN-Projekt „Entwicklung und Demonstration fortgeschrittener Online-Präsentationstechnologien“ [12].

Literatur

1. Kunze, Axel: Know-how zu Streaming-Video-Servern/Digitales Daumenkino, iX 10/1997 S.142-148
2. Loviscach, Jörn Dr.: Privatfernsehen/Video-Übertragung per Internet, c’t 6/2000 S.154-166
3. Real-System-Anleitung: http://service.real.com/help/library/guides/productiong27/realpgd.htm
4. QuickTime-Anleitung: http://www.apple.com/quicktime/authoring/tutorials.html
5. Windows-Media-Anleitung: http://msdn.microsoft.com/windowsmedia/
6. Apple MakeRefMovie: http://developer.apple.com/quicktime/quicktimeintro/tools/
7. Kobylinska, Anna; Martins, Filipe: Live-Video im Internet mit QuickTime 4.0, iX 8/1999 S.73-75
8. Rutledge, Lloyd: SMIL: Synchronized Multimedia Integration Language/Tonangebend, iX 10/1999 S.58-63
9. Open Source Darwin Streaming Server: http://www.publicsource.apple.com/projects/streaming/
10. Shout Interactive: http://www.shout3d.com
11. Pulse Entertainment: http://www.pulse3d.com
12. DFN-Projekt „Entwicklung und Demonstration fortgeschrittener Online-Präsentationstechnologien“: http://www.dfn-expo.de/Technologie/Streaming/Streaming_01.html
13. World Wide Web Consortium: HTML 4.01 Specification, http://www.w3.org/TR/html4/
14. World Wide Web Consortium: Synchronized Multimedia, http://www.w3.org/AudioVideo/
15. Bouthillier, Larry: Synchronized Multimedia On The Web (Tutorial zum Erlernen von SMIL), http://www.webtechniques.com/archives/1998/09/bouthillier/
16. Rau, Andreas; Renner, Frank: Video- und Audioübertragungen im Internet - Strom der Zeit, iX 10/1999 S.70-74
17. Einbetten von Real Movies in HTML-Seiten: http://service.real.com/help/library/guides/production/htmfiles/control.htm
18. Einbetten von QuickTime-Objekten in HTML-Seiten: http://www.apple.com/quicktime/
19. RZ-Kolloquium vom 28.03.2000:
    http://www.hu-berlin.de/rz/multimedia/bereiche/video/streaming/barz/stream1.htm
    http://www.hu-berlin.de/rz/multimedia/bereiche/video/streaming/lanyi/

Bernhard Barz
barz@rz.hu-berlin.de

Katrin Lanyi
katrin.lanyi@rz.hu-berlin.de