

# SMIL und RTP/RTSP

Torsten Lenhart

# Einführung

- Internet wurde für Austausch textueller Inhalte geschaffen
- Idee: Nutzung auch als Plattform für multimediale Kommunikation



Problem: herkömmliche Techniken und Protokolle können benötigte Funktionalität nicht bieten

- kein Standard für Synchronisation von Medien
- TCP/IP-Protokollfamilie nicht auf Echtzeitkommunikation ausgelegt

# Einführung (2)

- Lösungen:
  - **Synchronized Multimedia Integration Language (SMIL)** als Standard für Mediensynchronisation
  - **Real-Time Transport Protocol (RTP)** als Protokoll zur Übertragung echtzeitsensitiver Daten
  - **Real-Time Streaming Protocol (RTSP)** zur Steuerung und Kontrolle von Medienströmen

# Synchronized Multimedia Integration Language (SMIL)

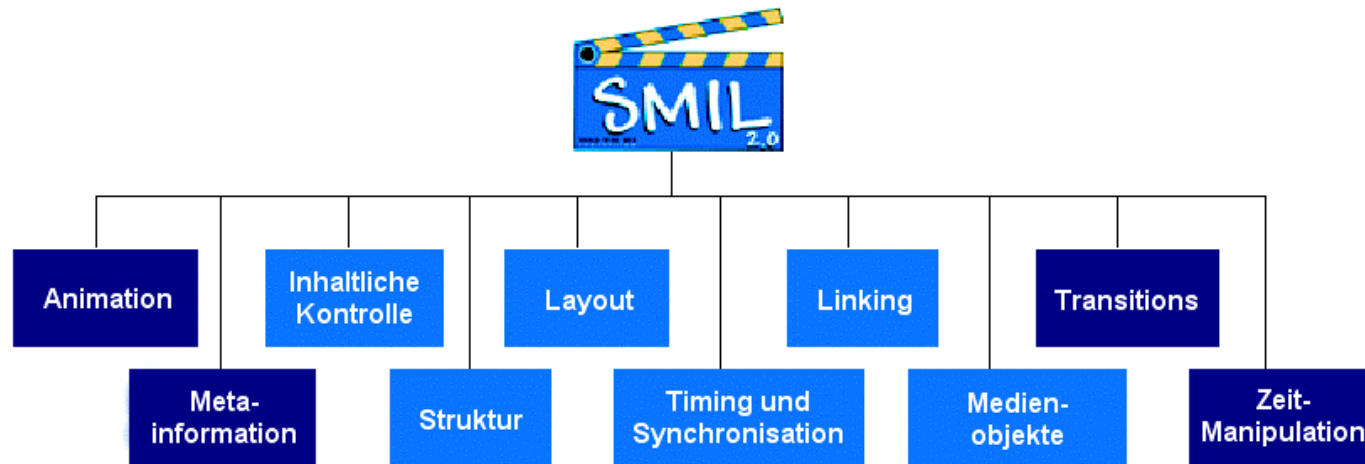
- Sprache zur Synchronisation und Integration unterschiedlicher Medienobjekte
- basiert auf XML
- kompatibel zu anderen XML-basierten Standards
- enge Verwandtschaft zu HTML
- keine Authoring-Tools notwendig
- leicht erlernbar

# Entwicklungsgeschichte

- *Dezember 1995*: Beginn der Entwicklung
- *November 1997*: SMIL 1.0
- *März 1998*: Erste Implementierungen in Software
- *August 1999*: Beginn der Arbeiten an SMIL 2.0
- *August 2001*: SMIL 2.0

# Modulgruppen

- SMIL ist eine Kollektion von XML-Elementen und Attributen
- Elemente und Attribute werden in Modulen und diese wiederum in Modulgruppen zusammengefasst
- Modulgruppen in SMIL:



# Modulgruppe „Struktur“

- **Elemente:**
  - `smil`: Definiert ein SMIL-Dokument, enthält alle anderen Elemente
  - `head`: enthält Meta- und Layout-Informationen
  - `body`: beinhaltet die restlichen Informationen

- **Beispiel:**

```
<smil>  
  <head>  
    ...  
  </head>  
  <body>  
    ...  
  </body>  
</smil>
```

# Modulgruppe „Medienobjekte“

- Wichtige Elemente:
  - Je ein Element für verschiedene Medientypen: `animation`, `audio`, `img`, `text`, `textstream`, `video` und `ref`
- Wichtiges Attribut für diese Elemente:
  - `src`: URI, um auf Medienobjekt zuzugreifen

- Beispiel:

```

```

```
<audio src="http://www.w3c.org/SYMM/joe-audio.wav" ... />
```

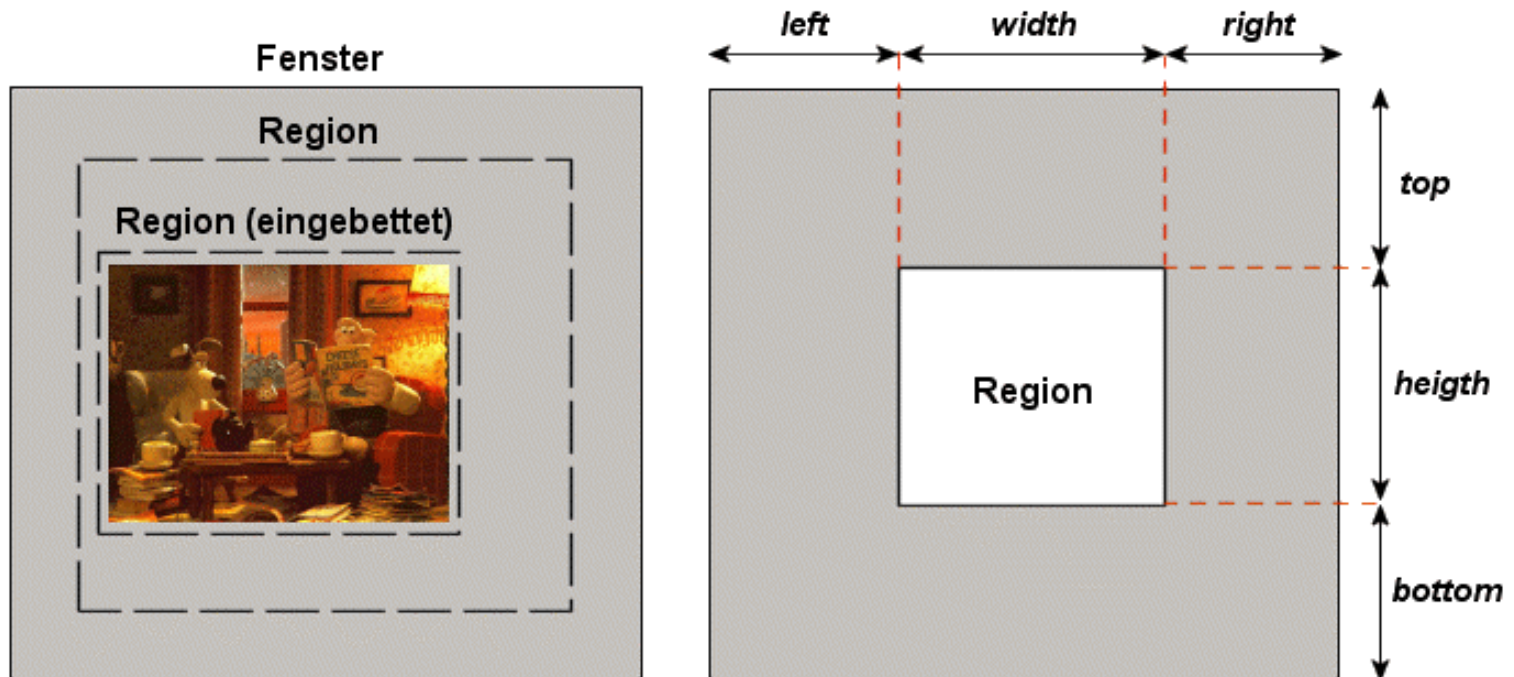
```
<video src="rtsp://www.cwi.nl/SMIL/video.rm" id="film"... />
```



# Modulgruppe „Layout“

- Hauptelement `layout` beinhaltet:
  - Element `root-Layout`:
    - repräsentiert ein Präsentationsfenster
    - Attribute `title`, `width`, `height` und `backgroundColor`
  - Element `region`:
    - repräsentiert Bereich innerhalb eines Präsentationsfensters
    - kann geschachtelt werden
    - ausser `title` die selben Attribute wie `root-Layout`, zusätzlich `top`, `bottom`, `left`, `right` und `regionName`
- Zusätzliches Attribut `region` für Medienobjekte, dient in Verbindung mit `regionName` zur Zuordnung zu einer Region

# Modulgruppe „Layout“ (2)



# Modulgruppe „Layout“ (3)

- Beispiel:

...

```
<layout>
```

```
  <root-Layout title="Beispiel" width="800"  
    height="600"/>
```

```
  <region regionName="Region"
```

```
    top="40" left="40" width="760" height="560">
```

```
    <region regionName="eingebetteteRegion"
```

```
      top="130" left="4" width="460" height="300"/>
```

```
    </region>
```

```
</layout>
```

...

```

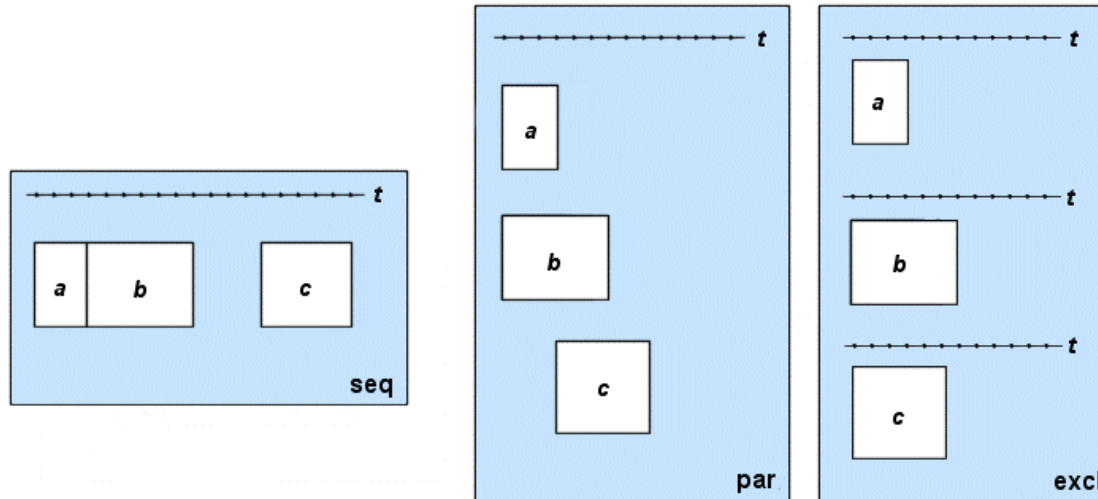
```

...

# Modulgruppe „Timing und Synchronisation“

- Kern der SMIL-Spezifikation
- drei Hauptelemente:
  - `seq`: sequentielle Darstellung von Medienobjekten
  - `par`: parallele Darstellung von Medienobjekten
  - `excl`: nur ein Medienobjekt gleichzeitig dargestellt, vorweg keine Reihenfolge vorgegeben, Zuteilung der Darstellungserlaubnis hängt von eingetretenen Ereignissen ab
- **Attributerweiterungen für Medienobjekte:** `begin`, `dur`, `end` und `repeat`

# Modulgruppe „Timing und Synchronisation“ (2)



```
<seq>  
    
    
    
</seq>
```

# Modulgruppe „Inhaltliche Steuerung“

- Mit Hilfe des Elements `switch` kann in Verbindung mit sog. Testattributen eine dynamischen Anpassung der Darstellung an die Systemumgebung erfolgen
- Beispiel mit Testattribut `systemBitrate`:

```
<switch>
  <video src="rtsp://www.cwi.nl/SMIL/video.rm"
    systemBitrate="115200" />
  <seq systemBitrate="57344">
    
    ...
    
  </seq>
</switch>
```

# Modulgruppe „Linking“

- ähnliches Linking-Konzept wie in HTML
- Erweiterung um zeitliche Vernetzung:
  - in Bezug auf Ziel: Es kann an eine bestimmte Stelle innerhalb einer Präsentation gesprungen werden
  - in Bezug auf Ausgangspunkt: Es kann in einem bestimmten Zeitabschnitt der Präsentation und nur dort ein Link existieren

# SMIL-Profile

- bieten für verschiedene Anwendungsformen einen optimalen und angepassten Sprachumfang
- vordefinierte Profile:
  - *SMIL 2.0 Language Profile (SMIL Profile)*: gesamter Sprachumfang
  - *SMIL 2.0 Basic Language Profile (SMIL Basic)*: speziell für mobile Endgeräte reduzierter Sprachumfang
  - *XHTML+SMIL*: Integration von SMIL in XHTML
  - *SMIL 1.0*: Sprachumfang der Vorgängerversion



# Zusammenfassung SMIL

- SMIL bietet umfangreiche Möglichkeiten zur Entwicklung von Multimedia-Präsentationen
- leichte Einsetz- und Erlernbarkeit
- mächtiges Timing-Konzept
- breite Integrations- und Wiederverwendungsmöglichkeiten
- unabhängig in Bezug auf Plattform, Formate und Art des Datentransports (verwendete Protokolle)

# Real-Time Transport Protocol (RTP)

- anwendungsintegriertes Protokoll zum Transport von echtzeitabhängigen Inhalten
- in der Regel Multicast, aber auch Unicast-Verbindungen möglich
- wird oft zur Realisierung von Audio- und Audio/Video-Konferenzen über das Internet eingesetzt
- seit November 1995 geprüfter Internetstandard (RFC 1889)

# Warum braucht man RTP?

- Alternative: Transportprotokolle der TCP/IP-Protokollfamilie benutzen:
  - TCP:
    - verlustfrei → führt im Lastfall zu massiven Problemen
    - schlechte Multicast-Unterstützung
  - UDP:
    - ausgeprägte Fehlererkennungsdienste
    - Multicast leicht zu realisieren
    - aber: keine Sequenznummernvergabe
- durch eigenes Protokoll können weitere interessante Funktionalitäten realisiert werden

# Merkmale

- für jedes übertragene Medium wird eigene RTP-Sitzung eingerichtet
- Vergabe von Sequenznummern und Zeitstempeln
- RTP wird von jeder Anwendung separat implementiert
- Erweiterungen, Modifikationen und Festlegung von Nutzdatenformaten in einem *RTP-Profil*
- **Achtung:**
  - keine Garantie, dass ein Paket rechtzeitig ankommt
  - keine Maßnahmen zur Flusskontrolle und Stauvermeidung

➔ Dienstgüte hängt von den Diensten der unteren Schichten ab

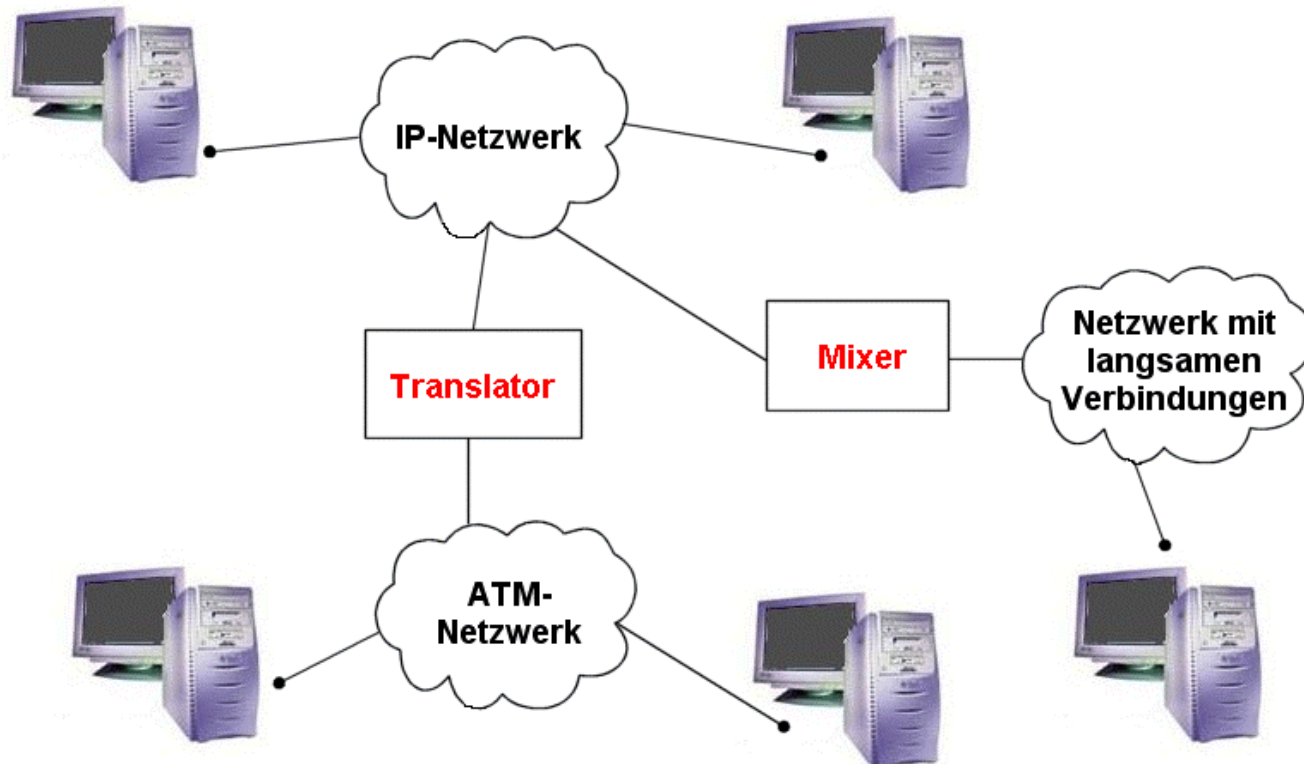
# RTP Control Protocol (RTCP)

- Hilfsprotokoll im Rahmen von RTP
- Aufgabe: Kanonisches Senden von Kontrollpaketen und Auswertung der Ergebnisse
- dadurch:
  - Austausch von Informationen über die Teilnehmer einer RTP-Sitzung
  - Bereitstellung von Informationen über die zur Verfügung stehende Dienstgüte

# Mixer und Translator

- zwischengeschaltete Geräte zur Unterstützung des Ablaufs einer RTP-Sitzung
- Mixer:
  - empfängt Pakete mehrerer Quellen (z.B. Audioströme)
  - rekonstruiert Timing und Sendereihenfolge
  - mischt alles zu einem Ausgabestrom zusammen
  - dabei eventuell Anpassung an die verfügbare Bandbreite auf der Ausgabeleitung
- Translator:
  - verbindet Subnetze mit unterschiedlicher Protokollstruktur
  - dient zur Überbrückung einer Application-Level Firewall
  - kann Ver- und Entschlüsselungstechniken anwenden

## Mixer und Translator (2)



# Zusammenfassung RTP

- Protokoll zum Transport echtzeitsensitiver Daten
- sehr flexibel wegen Anwendungsintegration und der Möglichkeit, angepasste RTP-Profile zu erstellen
- breite Anwendbarkeit durch Mixer und Translator
- mittlerweile oft verwendet, speziell zur Realisierung von Online-Konferenzen



# Real-Time Streaming Protokoll (RTSP)

- Protokoll zur Steuerung von Strömen kontinuierlicher Medien
- kein Content, d.h. die Daten selbst werden über ein separates Protokoll übertragen
  - ➔ oftmals RTP, aber auch andere Transportprotokolle
- seit April 1998 geprüfter Internetstandard (RFC 2326)

# Merkmale

- mit Hilfe von RTSP wird bestimmt, wann die Übertragung eines Medienstroms gestartet, pausiert oder beendet wird
  - ➔ fungiert quasi als „Netzwerk-Fernbedienung“
- Begriffe:
  - Präsentation: logisch zusammengehörige Ströme (z.B. Audio- und korrespondierender Videostrom)
  - Konferenz: Präsentation mit mehreren Clients und Servern
- für jede Präsentation existiert ein *Presentation Description File* mit Informationen über die Präsentation selbst und Angaben über die übertragenen Medien

## Merkmale (2)

- drei Verwendungsszenarien:
  - Kontrolle der Übertragung von kontinuierlichen Mediendaten
  - Einladung eines Medienservers zu einer Konferenz
  - Information über Eintreffen zusätzlicher Daten
- RTSP ist für Medienströme das, was HTTP für HTML-Dokumente ist
- Unterschiede:
  - RTSP ist nicht zustandslos
  - sowohl Client als auch Server sind berechtigt, Anfragen abzusetzen

## Merkmale (3)

- zur Realisierung der Dienste sind Methoden definiert, z.B.
  - SETUP: Client fordert Verbindung beim Server an
  - PLAY: Client fordert Server zum Senden über zuvor eingerichtete Verbindung auf
  - PAUSE: kurzzeitige Unterbrechung ohne Ressourcenfreigabe
  - TEARDOWN: beenden der Verbindung und Freigabe der Ressourcen
  - DESCRIBE: Beschreibung eines Medienstroms wird an Client gesendet

# Zusammenfassung RTSP

- Protokoll zur Kontrolle von Medienströmen
- breite Anwendbarkeit wegen Unabhängigkeit von Protokollen der unteren Schichten
- große Palette an Methoden zur Steuerung
- Ähnlichkeit mit HTTP, dadurch schneller Einstieg in den Umgang, wenn HTTP bereits bekannt ist

# Zusammenfassung

- Vorgestellte Techniken bieten Möglichkeit, die Lücke zwischen vorhandener und benötigter Funktionalität in Bezug auf Multimedia-Fähigkeit des Internets zu schließen

Mit **RTP** können multimediale, echtzeitsensitive Daten adäquat übertragen werden

- ➔ die entsprechenden Ströme sind durch **RTSP** bequem steuerbar
- ➔ und können schließlich durch **SMIL** im Rahmen einer Multimedia-Präsentation synchronisiert und integriert dargestellt werden