MPEG-4 Workshop, AES 106th Convention,
Munich, Germany, May 10, 1999

AudioBIFS:
Audio Composition in MPEG-4 version 1

Jyri Huopaniemi

Nokia Research Center

Speech and Audio Systems Laboratory

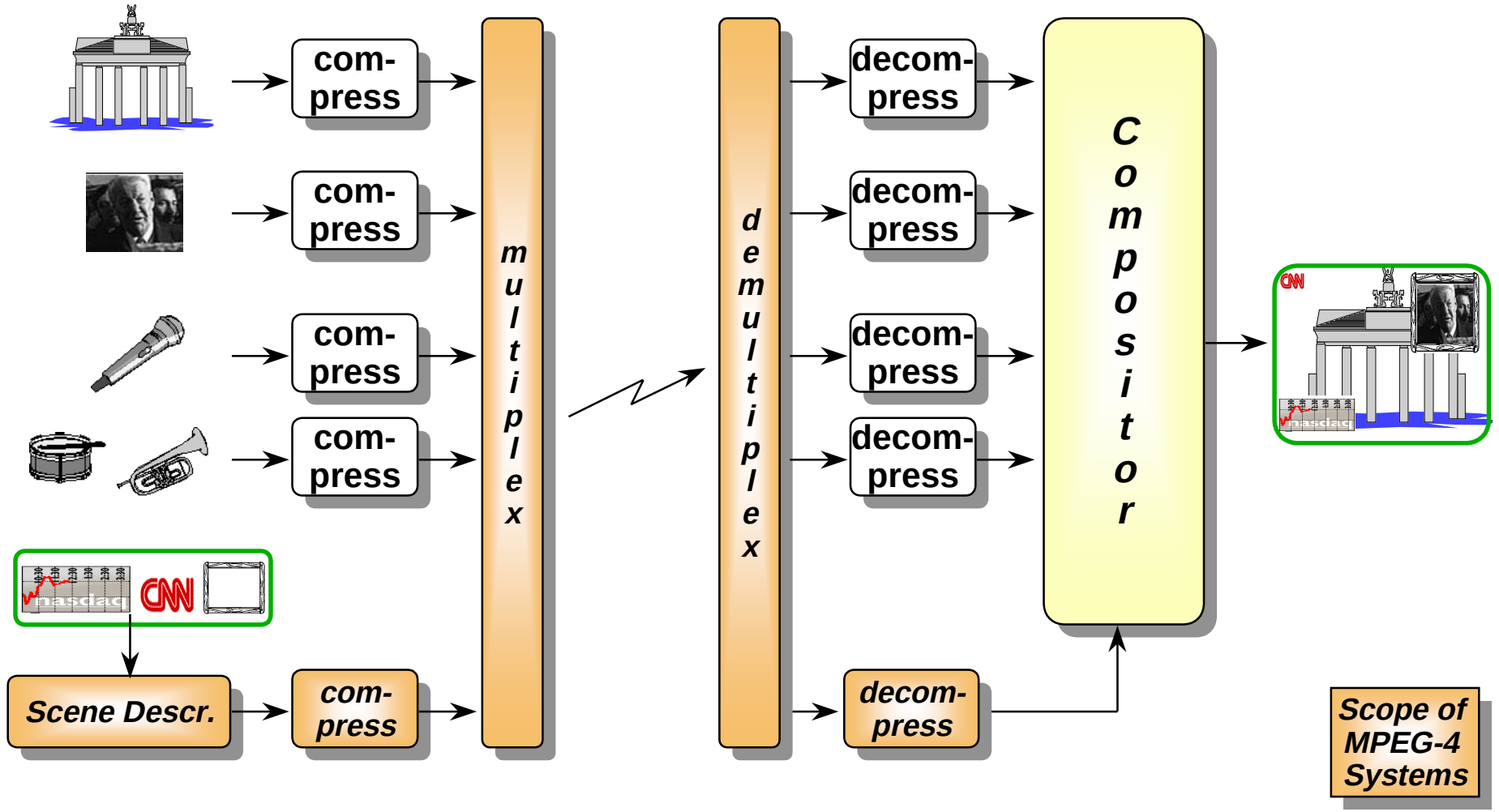
P.O.Box 407, FIN-00045 NOKIA GROUP, Finland

jyri. huopaniemi@nokia.com

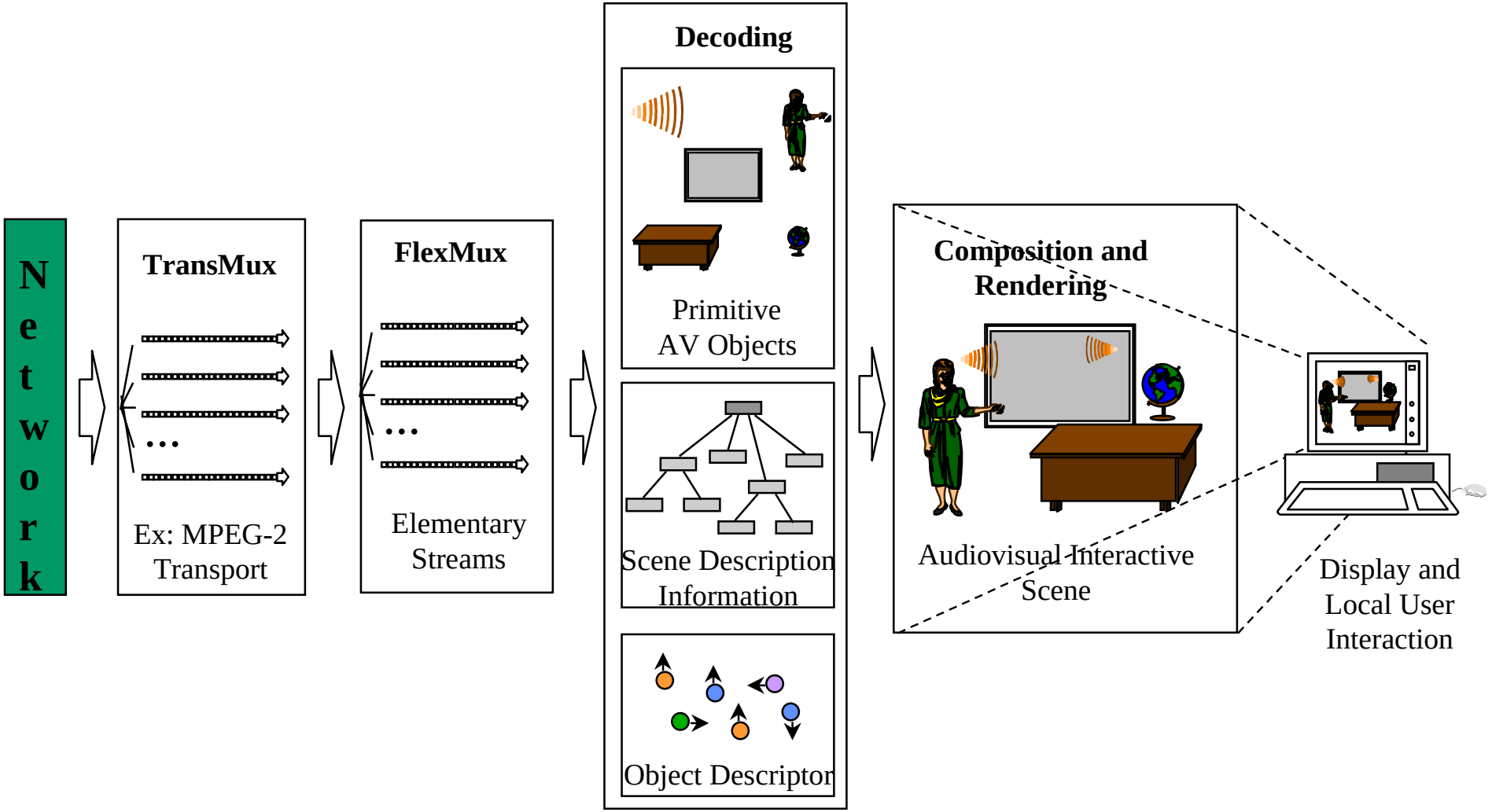
Acknowledgements

- Presentation based on:
  - MPEG-4 v.1 FDIS
  - Eric Scheirer, Riitta Väänänen, and Jyri Huopaniemi, "AudioBIFS: The MPEG-4 Standard for Effects Processing", *Proc. DAFX98 Workshop on Digital Audio Effects*, Barcelona, Nov. 1998.
  - Eric Scheirer, Riitta Väänänen, and Jyri Huopaniemi, "AudioBIFS: Describing Audio Scenes with the MPEG-4 Multimedia Standard", Accepted for publication in: *IEEE Transactions on Multimedia*, 1999.
- Special thanks to my co-authors Eric Scheirer (MIT Media Lab) and Riitta Väänänen (Helsinki Univ. of Tech., Acoustics Lab)
  - Eric for providing the audio demo
  - Riitta for providing the IM-1 audio scene description demo

MPEG-4 Systems Structure



MPEG-4 Systems Decoder Structure



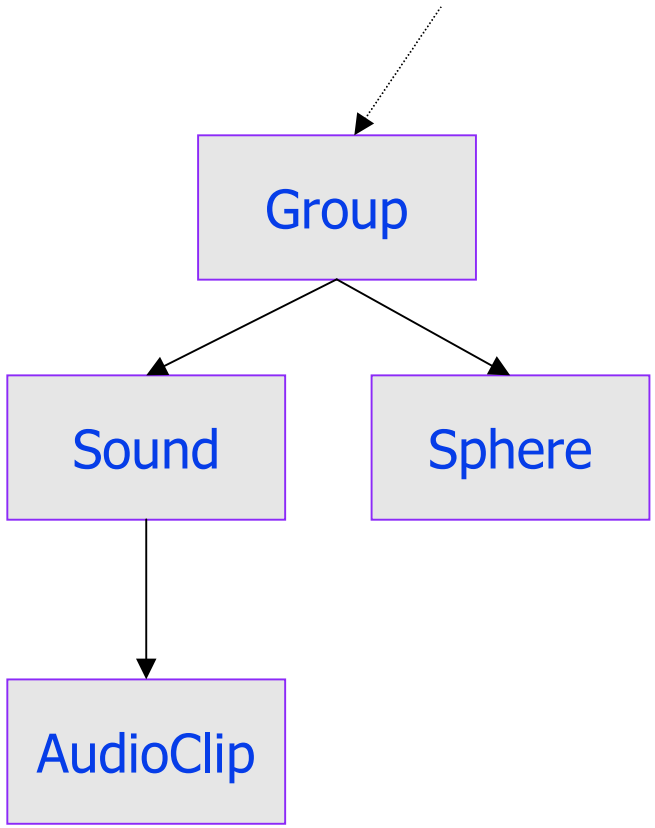
MPEG-4 BIFS

- Scene description = Coded representation of interactive audiovisual scene
  - *Associate elementary streams with scene description*
- MPEG-4 compressed *BInary Format for Scenes* (BIFS)
- Many concepts inherited from VRML
- Many new concepts in MPEG-4
  - Streaming media with synchronization
  - Dynamic scene changes (BIFS-Anim)
  - Geometry compression
  - 2D and 2D/3D scenes
  - Rich model for audio mixing & post-production

MPEG-4 v.1 Audio Composition

- Starting point: Virtual Reality Modeling Language (VRML) standard (ISO IEC JTC/SC24 IS 14772-1, 1997)
- All VRML (& BIFS) based on *scene graph*
  - Media content introduced and represented as a set of *Nodes*
  - *Nodes* interconnected in hierarchical relationships
- Two sources for audio in VRML
  - **AudioClip** contains WAVE or MIDI data
  - **MovieTexture** has audio synchronized to video clip
- Sound attached to objects
  - **Sound** node used to put sound in the scene
  - “Elliptical” model of attenuation

Audio in VRML: Example



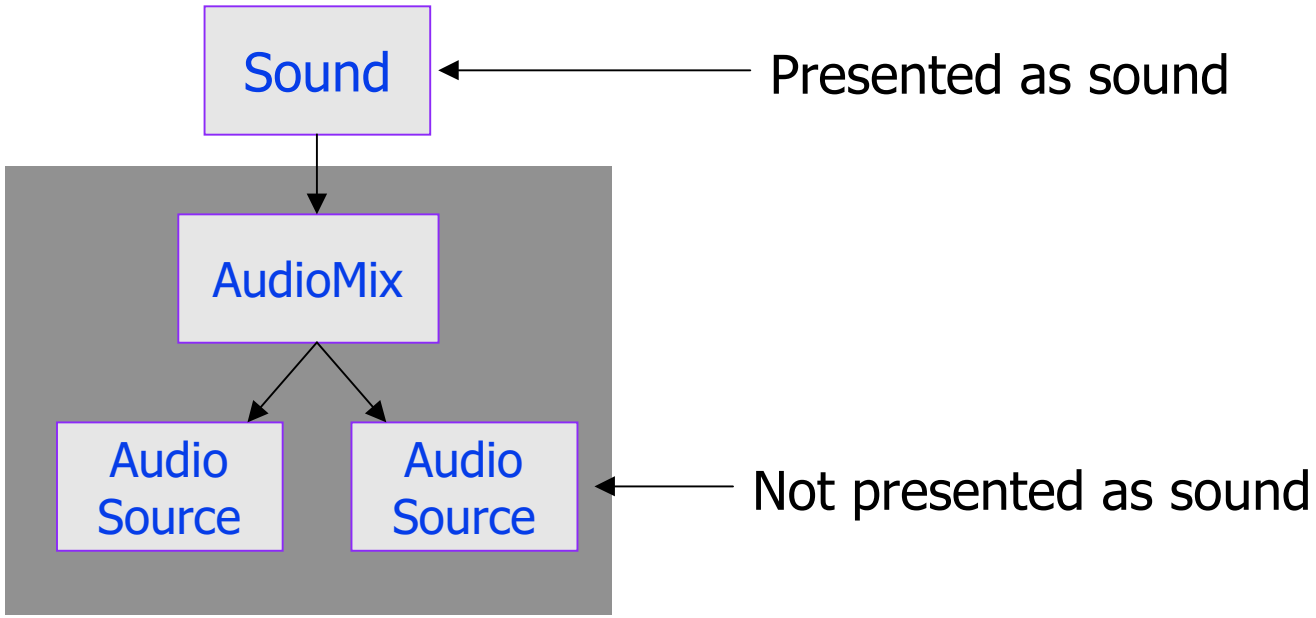

```
Group {
  children [
    Sphere {
      radius 3
      position 0,0,0
    }
    Sound {
      position 0,0,0
      source
        AudioClip {
          url “http://…/file.wav”
          loop = 1;
        }
    }
  ]
}
```

MPEG-4: Two sound modeling strategies

- *Virtual-reality compositing*
  - Goal: physically recreate a sound scene (like VRML)
  - Attach sound to objects, position in 3-D space
- *Abstract-effects compositing*
  - Goal: provide tools for artists
  - Download multiple sounds
  - Use signal-processing to mix / produce effects etc.
- Use either technique (or both) in MPEG-4

MPEG-4 Audio Scene Graphs

- **Sound** node in BIFS attaches an "audio scene graph"
- Nodes below **Sound** create one "object"



MPEG-4 Version1 AudioBIFS Nodes

Node	Purpose	
AudioBuffer	Insert an audio clip to scene	
AudioClip	Interactively trigger snippets of sound	**from VRML**
AudioDelay	Insert delay to sound	
AudioMix	Mix sounds	
AudioSource	Define audio source input to scene	
AudioFX	Attach structured audio objects to sound	
AudioSwitch	Switching of audio sources in scene	
Group, Group2D	Grouping of nodes and subtrees in a scene	
ListeningPoint	Define listening point in a scene	
Sound	Define properties of sound	**from VRML**

AudioBIFS nodes

- **AudioSource**
  - Provides streaming audio to the scene
  - Attaches to any MPEG-4 audio decoder

- **AudioMix**
  - Mix M channels of sound into N channels
  - Mixing matrix can be controlled interactively

AudioBIFS nodes

- **AudioDelay**
  - Delay sounds slightly, for synchronization
- **AudioSwitch**
  - Choose N channels out of a set of M
  - (AudioMix with 1/0 matrix, N < M)
- **AudioBuffer**
  - Grab a “clip” for interactive use

    (Also used to compress samples for wavetable synthesis)

AudioBIFS nodes

- **AudioFX**
  - Download custom effects algorithms
  - Apply to M channels to get N channels
  - SAOL is used to describe effects

    (integrated with Structured Audio toolset)

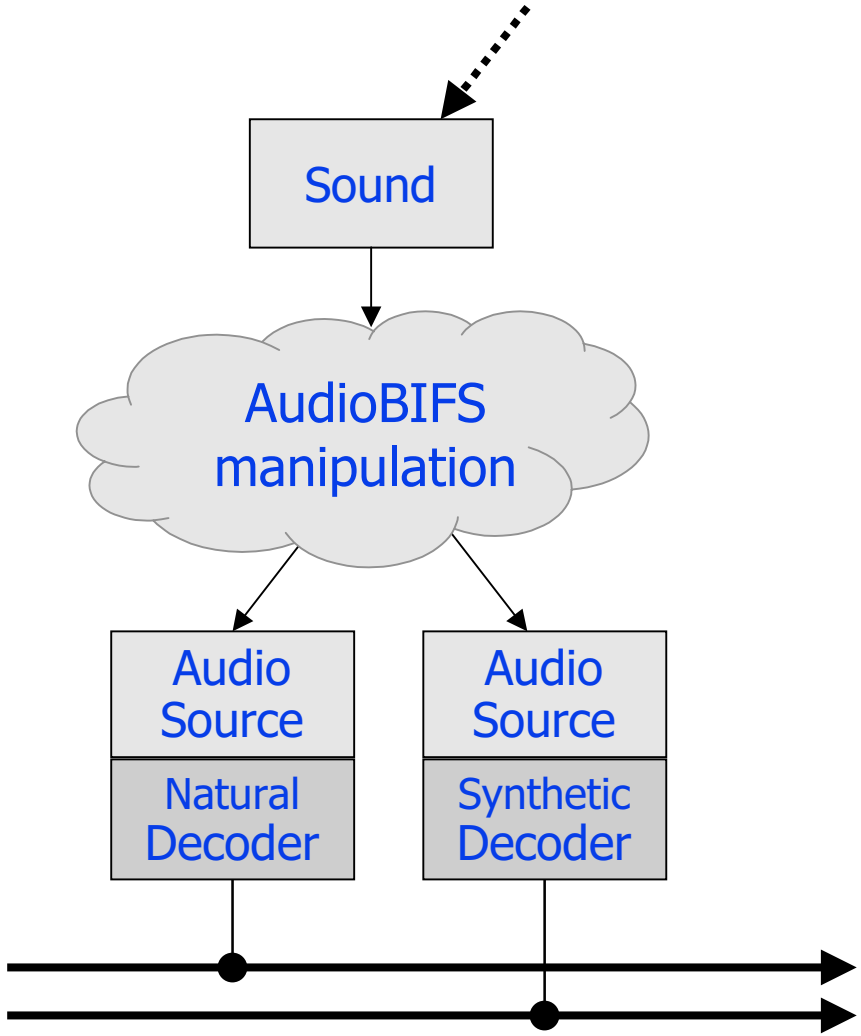
  - Important philosophy:
  - MPEG didn’t pick “the standard reverb”

AudioBIFS nodes

- **Sound**
  - Just like VRML, slightly more detailed semantics
- **Sound2D**
  - Put sound in 2-D scenes
- **ListeningPoint**
  - Control the “virtual ear” in the scene
- **TermCap**
  - Make scenes change depending on local resources

    (Number of speakers, dynamic range, max sampling rate)

The big picture



Attach sound to main scene (spatially position if desired)

Create sound object with AudioBIFS

Inject sound into scene graph

Decode into raw audio samples

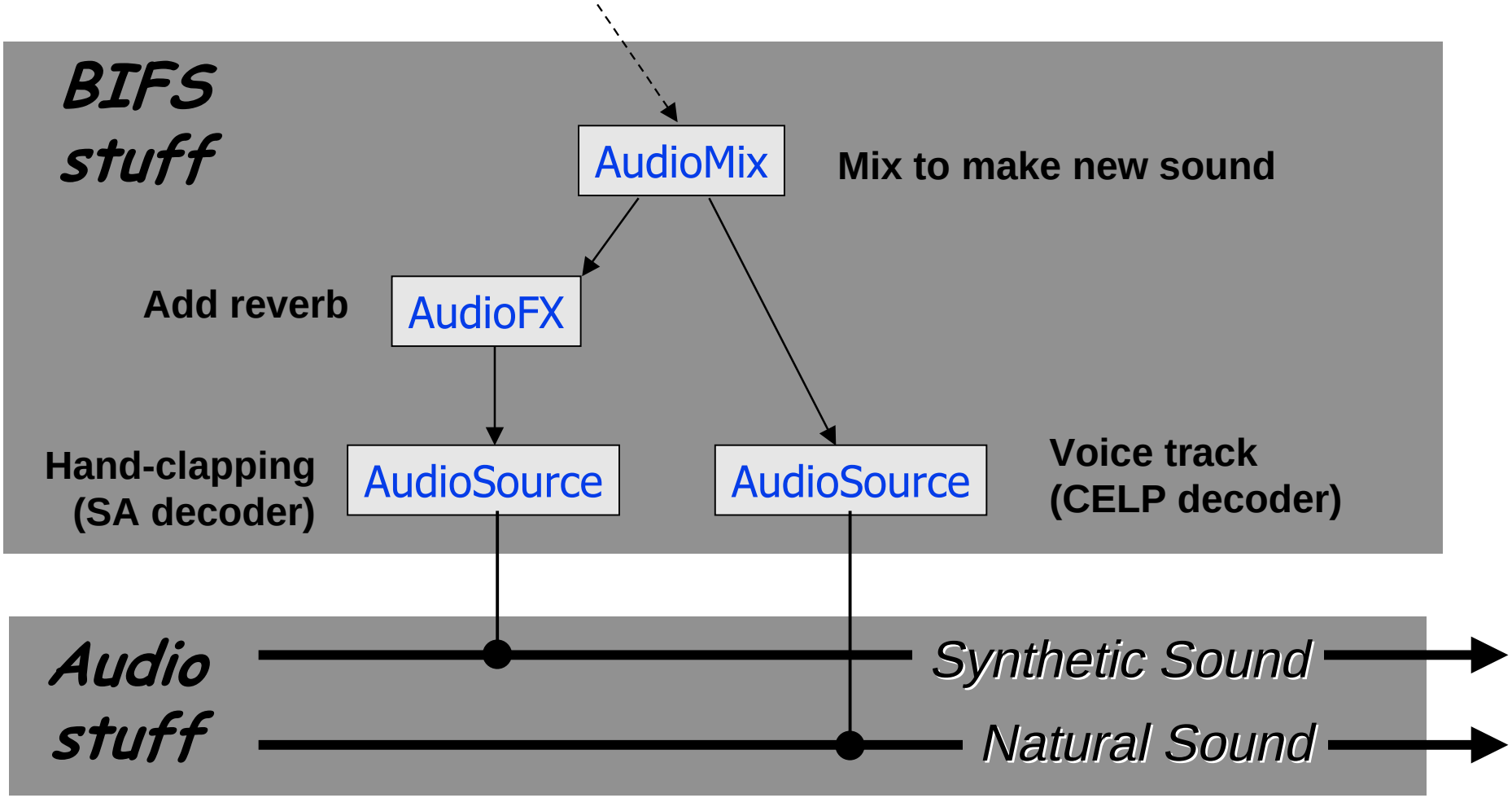
Streaming compressed audio

Demonstration

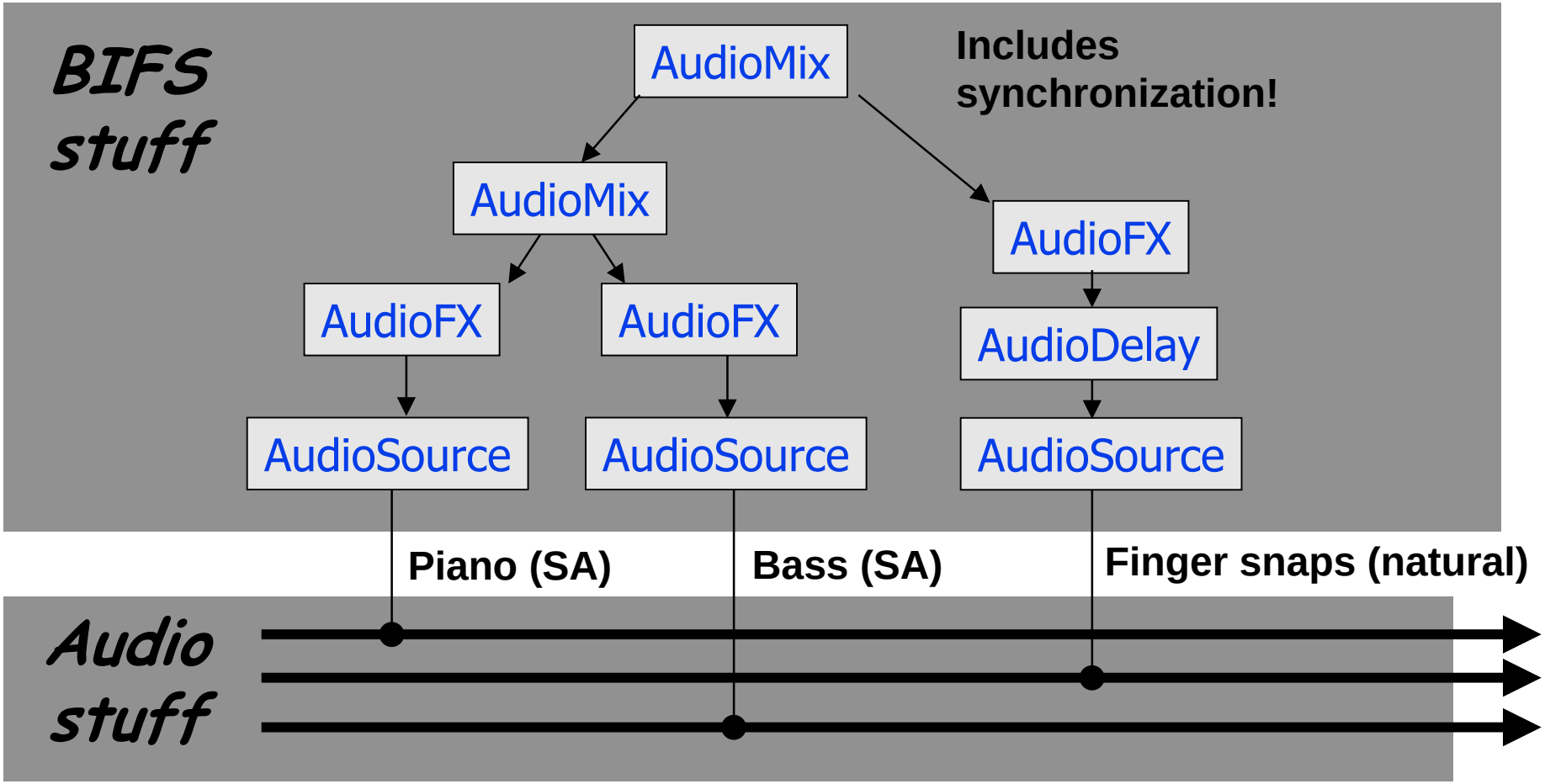
- **16 kbps** MPEG-4 soundtrack

- Natural and synthetic speech
- Synthetic music and sound effects
- Mixdown in AudioBIFS
- Effects-processing described in SAOL

Structured audio scene



Structured audio scene



Demonstration

- MPEG-4 Reference Software
  - 3-D IM-1 player
- Examples of MPEG-4 v.1 Audio scene description
  - Polygon model of a concert hall with a sound source (coded audio file)
  - Simple room model with two human speakers
  - Interactivity, 3-D positioning, distance effects

MPEG-4 version 2: Advanced AudioBIFS

- More advanced “virtual reality” sound using physical modeling
  - Distance dependent attenuation
  - Directivity & modeling of sound propagation
  - Air absorption, Doppler effect
  - Modeling of room reflections and transmission
  - Parameters for late reverberation
- More advanced audio postprocessing using perceptual modeling
  - Perceptual parameters for controlling sound sources and rooms independently of the visual scene
- To be completed early 2000, currently in Committee Draft stage

Conclusions

- MPEG-4 AudioBIFS provides a powerful unifying framework for:
  - Virtual-reality audio scenes
  - Abstract-effects compositing for “serious content”
  - Downloadable effects algorithms
  - Streaming and interactive sound
  - Environmental modeling (version 2)
- For more information & code:
  - <http://sound.media.mit.edu/mpeg4>