CS 414 – Multimedia Systems Design

# Lecture 34 – Synchronization (Part 2)

Klara Nahrstedt

Spring 2008

# Administrative

- MP4 posted
  - April 29 (preview for finalists)  4-6pm
  - April 30 grading for non-finalists 3:30-5pm and competition for finalists  5-7pm
- Peer Evaluation
  - Deadline May 5, 5pm – very important 5% of your grade
  - See website posting
- Discussion Session
  - April 15 (Tuesday), 7pm, 3401 SC

# Outline

- Synchronization Reference Models
- Synchronization in Distributed Environments
- Location of Synchronization
- Clock Synchronization

# Reference Models

- We need reference models to
  - Understand various requirements for multimedia sync
  - Identify and structure run-time mechanisms to support execution of sync
  - Identify interface between run-time mechanisms
  - Compare system solutions for multimedia sync

# Existing Models

- Little and Ghafoor
  - Sync multimedia objects are classified according to inter-media and intra-media sync, live and synthetic sync at levels:
    - **(a) Human level; (b) System level; (c ) Physical level**
- Ehley, Furth, Ilyas
  - Sync multimedia objects are classified according to control jitter between media streams and with respect to distributed sync control:
    - **(a) using protocols, (b) using servers, (c ) using nodes without server structure**

# Synchronization Reference Model

- Sync model we will be evaluating in detail is according to Meyer, Effelsberg, Steinmetz:
  - Sync multimedia objects are classified according to
    - Media level
    - Stream level
    - Object level
    - Specification level

# Media Level (1)

- Each application operates single continuous media streams composed of sequence of LDUs
- Assumption at this level: device independence
- Supported operations at this level:
  - *read(devicehandle, LDU)*
  - *write(devicehandle, LDU)*

# Media Level (2) - Example

```
window = open(“videodevice”);
movie = open(“file”);
while (not EOF (movie) ) {
        read(movie, &LDU);
        if (LDU.time == 20)
                printf(“Subtitle 1”);
        else if (LDU.time == 26)
                printf(“Subtitle2”);
        write(window, LDU);  }
close(window);
close(movie);
```

# Stream Level (1)

- Operates on continuous media streams and groups of streams
- Models inter-stream synchronization for need of parallel presentation
- Offers abstractions:
  - notion of streams,
  - timing parameters concerning QoS for intra-stream and inter-stream synchronization

# Stream Level (2)

- Supports operations:
  - Start(stream), stop(stream), create-group(list-of-streams);
  - Start(group), stop(group);
  - Setcuepoint(stream/group, at, event);
- Classifies implementation according to
  - Support for distribution (end-to-end, local)
  - Support of type of guarantees (best effort, deterministic)
  - Support of types of supported streams (analog, digital)

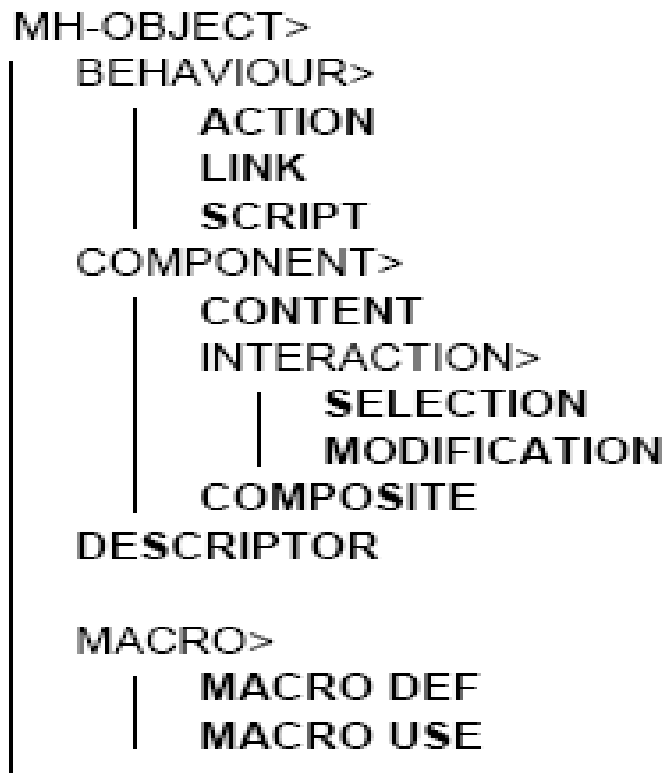
# Object Level (1)

- Operates on all types of media and hides differences between discrete and continuous media
- Offers abstractions:
  - Complete sync presentation
- Computes and executes complete presentation schedules that include presentation of non-continuous media objects and calls to stream level
- Does not handle intra-stream and inter-stream synchronization
  - (relies on media and stream levels)

# Object Level (2) - Example

- MHEG – Multimedia Hypermedia Experts Group of ISO
  - Defines representation and encoding of multimedia and hypermedia objects
  - Provides abstractions suited to real-time presentations
    - implemented via multimedia synchronization functionalities
  - Provides abstracts for real-time exchange
    - implemented with minimal buffering
  - Evaluates status of objects and performs actions (e.g., prepare, run, stop, destroy)
    - For time-dependent streams – access to stream level
    - For time-independent streams – direct access the object to present it
- Classification of this level according to (a) distribution capabilities, (b) type of presentation schedule, (c) schedule calculation

# MHEG Example (specified in SGML)




'>' means that this object has the following sub-classes.
Only the instances of the classes in bold
type may be interchanged.

# Specification Level

- Open layer included in tools which allow to create sync specifications
- Examples:
  - Synchronization editors, document editors, authoring systems, conversion tools
  - Examples of such tools: multimedia document formatter that produces MHEG specifications
- Classification:
  - Interval-based spec
  - Time-axes based spec
  - Control flow-based spec
  - Event-based spec

# Synchronization in Distributed Environments

- Information of synchronization must be transmitted with audio and video streams, so that receiver(s) can synchronize streams
- Sync information can be delivered before start of presentation (used by synthetic synchronization)
  - Advantage: simple implementation
  - Disadvantage: presentation delay
- Sync information can be delivered using separate sync channel - out-band (used by live synchronization)
  - Advantage: no additional presentation delay
  - Disadvantage: additional channel needed

# Sync in Distributed Environments

- Sync information can be delivered using multiplexed data streams - in-band sync
  - Advantage: related sync information is delivered together with media units
  - Disadvantage: difficult to use for multiple sources

# Location of Sync Operation

- Sync media objects by combining objects into new media object
- Sync operation placed at sink
  - Demand on bandwidth is larger because additional sync operations must be transported
- Sync operation placed at source
  - Demand on bandwidth smaller because streams are multiplexed according to sync requirements

# Other Sync Issues

- Sync must be considered during object acquisition
- Sync must be considered during retrieval
  - Sync access to frames of stored video
- Sync must be considered during transport
  - If possible use isochronous protocols
- Sync must be considered at sink
  - Sync delivery to output devices
- Sync must consider support of functions such as pause, forward, rewind with different speeds, direct access, stop or repeat

# Conclusion

| | |
|---|---|
| Specification Layer - tools | Editing and Formatting<br>Mapping of user-oriented QoS to abstractions at the object layer |
| Object Layer- Sync Spec. | Plan and Coordinate presentation<br>Initiate presentation of time-dep. media by the stream layer<br>Initiate presentation of time-indep. media<br>Initiate presentation preparation actions |
| Stream Layer | Resource reservation and scheduling |
| Media Layer | File and device access |