MPEG-4: Object Description, Synchronization and Scene Description

Presented by: Yongjie Liu

Mar. 28, 2003

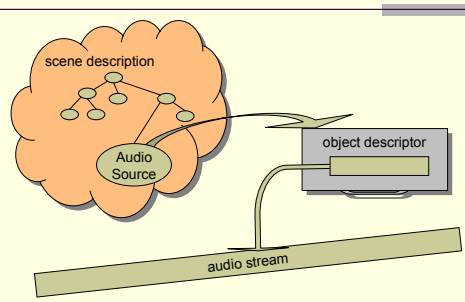
Outline

- Object Descriptions
- Semantic Description and Access Management
- Timing Model and Synchronization of Streams
- Introduction to Scene Description

MPEG-4 Content Description

- How to describe objects in MPEG-4?
  - OD framework
    - All components are described recursively through object descriptors (ODs)
    - Hierarchical structure (subdescriptors)
    - Example: `AudioSource`
- Scene description
  - Assemble objects into audiovisual scene
  - Scene description format—binary format for MPEG-4 scenes (BIFS)

MPEG-4 Content Description (cont)



Syntactic Description Language (SDL)

- A language expressing syntax of OD framework
  - Object-oriented structure, like C++
  - Variable definition

```
[aligned] type[(length)] element_name [=value];
```

Example:

```
aligned bit(16) picture_code=0x0100;
```

In bitstream, parse 16 bits from the next byte boundary and to check it to be hex value 100.

  - Class example

```
class foo {
  unsigned int(5) aVal;
  bit(3) bVal;
}
```


SDL (cont)

- Class tag

```
class ClassWithID : aligned bit(8) classTag=0x23 {
    Foobar aFoobar;
}
```

  - Tag value is parsed from bitstream before reading any element within the class.
  - Provide discrimination of different classes that may occur in bitstream at the same location
- Keyword “expandable”—limiting the length of class

```
abstract expandable (2^28-1) class ClassesWithID:aligned bit(8)
    classTag=0 {
       Foobar aFoobar;
}
```

Object Description

- All descriptors derived from a common base class—`BaseDescriptor`

Base Descriptor: The template for other descriptions

```
abstract aligned(8) expandable(2^28-1) class BaseDescriptor:bit(8) tag=0 {
        //empty, To be filled by classes extending this class.
}
```

- OD syntax

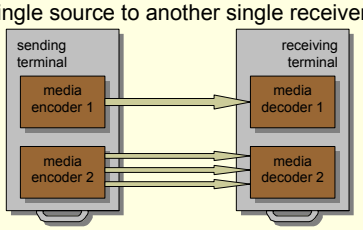
```
class ObjectDescriptor extends ObjectDescriptorBase: bit(8) tag=ObjectDescrTag {
    bit(10) ObjectDescriptorID;
    bit(1) URL_Flag;
    const bit(5) reserved=0b1111.1;
    if (URL_Flag) {
       bit(8) URLength;
       bit(8) URLstring(URLength);
    } else {
       ES_Descriptor esDescr[1..255];
       OCI_Descriptor ociDescr[0..255];
       IPMP_DescritporPointer ipmpDescriPtr[0..255];
    }
    ExtensionDescriptor extDescr[0..255];
}
```


Object Description (cont)

- Subdescriptors describe four elements:
  - Individual elementary stream (ESs)
  - Semantic information about an object (OCI—Object Content Information)
  - Hooks for content access management (IPMP—Intellectual property management and protection)
  - A placeholder for future extension descriptors

Stream Description

- Elementary stream is described by an elementary stream descriptor (ESD)
- ES—a flow of data that originates from a single source to another single receiver



ESD syntax

```
Class ES_Descriptor extends BaseDescriptor : bit(8) tag=ES_DescrTag {
    bit(16) ES_ID;
    bit(1) streamDependenceFlag;
    bit(1) URL_Flag;
    bit(1) OCRstreamFlag;
    bir(5) streamPriority;
    if (streamDependenceFlag)
        bit(16) dependsOn_ES_ID;
    if (URL_Flag) {
        bit(8) URLength;
        bit(8) URLstring(URLength);
    }
    if (OCRstreamFlag)
        bit(16) OCR_ES_Id;
    DecoderConfigDescriptor decConfigDescr;
    SLConfigDescriptor slConfigDescr;
    IPI_DescrPointer ipiPtr[0..1];
    IP_IdentificationDataSet ipIDS[0..255];
    IPMP_DescriptorPointer ipmpDescrPtr[0..255];
    LanguageDescriptor langDescr[0..255];
    RegistrationDescriptor regDescr[0..1];
    ExtensionDescriptor extDescr[0..255];
}
```


Decoder Configuration

Providing setup information required by the decoder to process stream

```
Class DecoderConfigDescriptor extends BaseDescriptor : bit(8)
                        tag=DecoderConfigDescrTag {
    bit(8) objectTypeIndication;
    bit(6) streamType;
    bit(1) upStream;
    const bit(1) reserved=1;
    bit(24) bufferSizeDB;
    bit(32) maxBitrate;
    bit(32) avgBitrate;
    DecoderSpecificInfo decSpecificInfo[0..1];
    profileLevelIndicationIndexDescriptor
                profileLevelIndicationIndexDescr[0..255];
}
```


Quality of Service Descriptor

- For media server and gateways to intelligently forward streams or sets of streams
- QoS Descriptor contains either an index of a predefined QoS scenario or a set of QoS qualifiers

```
class QoS_Descriptor extends BaseDescriptor : bit(8) tag=QoS_DescrTag {
    bit(8) predefined;
    if (predefined==0) {
       QoS_Qualifier qualifiers[];
    }
}

abstract aligned(8) expandable(2^28-1) class QoS_Qualifier : bit(8)
    tag=0x01..0xff {
    //empty. To be filled by classes extending this class.
}
```

Registration Descriptor

- Allow user to carry non-MPEG-4 ESs
- Registration Descriptor syntax

```
Class RegistrationDescriptor extends BaseDescriptor : bit(8)
            tag=RegistrationDescrTag {
   bit(32) formatIdentifier;
   bit(8) additionalIdentificationInfo[sizeOfInstance-4];
}
```

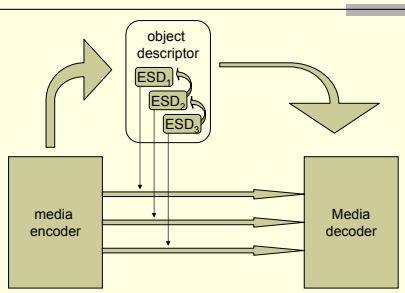
13

Stream Relationship Description

- Describing an object with a single ES descriptor consists in creating an ESD and placing it within an OD
- One OD may contain several ESDs (e.g. for different qualities)
- Scalable coding
  - Multiple streams, with base quality provided by first stream
  - Quality can be improved in multiple steps by concrete scalability algorithm, corresponding to the multiple ESs (enhancement layers)
  - Order to pass the set of streams is achieved by Stream dependency mechanism of ESD

14

Stream Relationship Description (cont)



Selecting a set of scalable streams

15

Content Complexity Descriptor

- A MPEG-4 terminal can decide whether to decode and present the content by content complexity
- Labels indicating profiles and levels for the content—*initial object descriptor* (IOD) and ***extension*** *profile level descriptor*
- Attaching multiple extension profile level descriptors to IOD to describe different presentations in one OD

16

Streaming ODs

- How to communicate ODs from sender to client terminal? (OD stream)
- MPEG-4 focus: enabling dynamic multimedia content
- Different from existing mechanisms for signaling stream properties. (SDP with RTSP and ITU-T: signaling stream only at beginning of presentation)

17

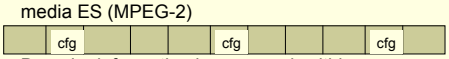
Stream ODs (cont)

- OD commands conveyed by OD stream
  - OD update
  - OD remove
  - ESD update
  - ESD remove
  - OD execute

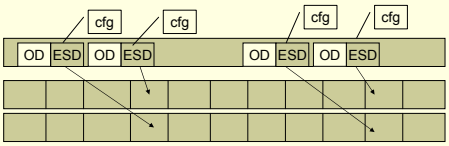
18

Streaming Configuration Information in ODs

- Decoder information is repeated regularly as part of media ES (MPEG-2)



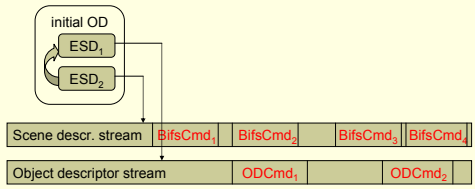
- Decoder information is conveyed within `DecoderSpecification` element in separate OD stream



19

Linking a Scene to its Media Streams

- OD streams are usually associated with a scene description (BIFS) stream by another OD or IOD

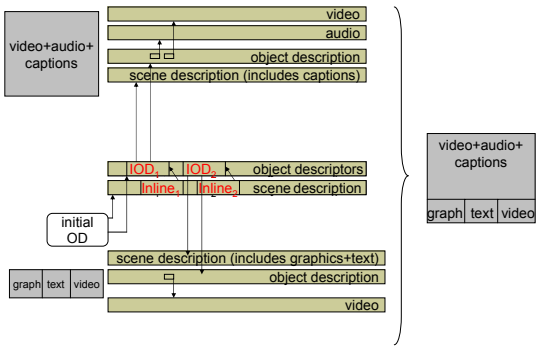


20

Association of MPEG-4 Presentation

- How to author small presentations and plug them together into bigger one?
- Use another scene description to inline the two scenes as subscenes. IODs of the subscenes will become part of the OD stream for the composite scene.

21

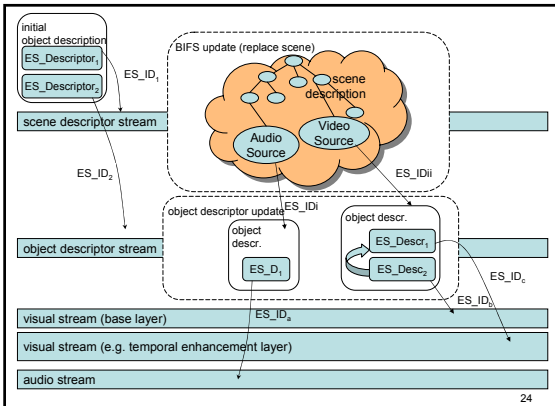


22

MPEG-4 Content Access Procedure

- Starting with an initial OD
- IOD contains pointers to at least two essential streams: a scene description (BIFS) and an OD stream.
- DMIF (Delivery Multimedia Integration Framework) Application Interface (DAI) as the API of the procedure, independently of the transport and signaling protocol
- Transporting the coded data—RTSP etc.

23



24

Outline

- Object Descriptions
- Semantic Description and Access Management
- Timing Model and Synchronization of Streams
- Introduction to Scene Description

25

Semantic Description and Access Management

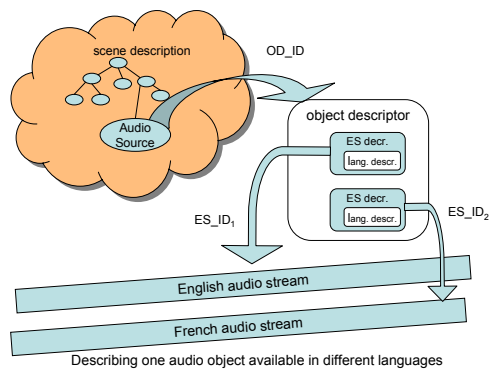
- Auxiliary streams providing semantic information and enabling access management
- OCI streams—providing metadata, describing objects semantically
- IPMP streams—providing hooks to control the legitimate usage of MPEG-4 content

26

Object Content Information: Meta Information About Objects

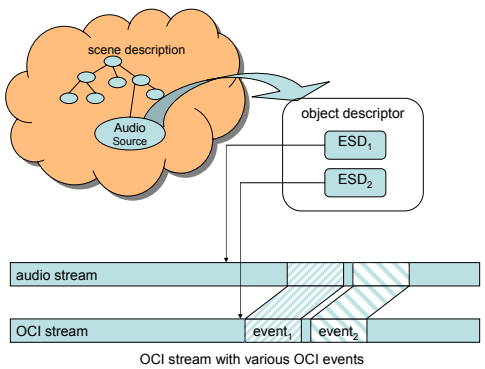
- A set of descriptors and a stream type have been defined to carry information about the media object in general: OCI descriptor and OCI stream
- OCI descriptors can be attached to OD, all ESs share same object content information
- OCI may be dynamically changing over time—a separate OCI stream is used to convey the information (OCI_Event message associates OCI descriptors with media stream)

27



Describing one audio object available in different languages

28



OCI stream with various OCI events

29

Intellectual Property Management and Protection

- IPMP framework to protect digital content of MPEG-4 (encrypt, watermark etc.)
- Intellectual property identification (IPI) data set, IPMP descriptors and IPMP streams
- IPI data set
  - `ContentIdentificationDescriptor`
  - `SupplementaryContentIdentificationDescriptor`
- IPMP descriptor and stream configure the IPMP system in terminal

30

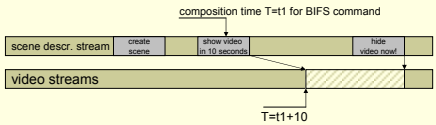
Outline

- Object Descriptions
- Semantic Description and Access Management
- Timing Model and Synchronization of Streams
- Introduction to Scene Description

31

Modeling Time

- Time in MPEG-4 is always relative.
- Finding a reference point.
  - Unicast: start of presentation playback
  - Multicast (or broadcast): the point in time when terminal receives scene description data (the point when BIFS command is due)



Time relative to composition time of a BIFS command

32

Time Stamps and Access Units

- Time stamps associated with streams define the relations between different streams
- Portions related to time in all stream type (OD stream, OCI, IPMP etc.) are called *access units* (AUs), where time stamps can be attached to.

elementary stream as a sequence of AUs

33

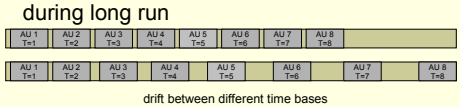
Packetizing Streams: The Sync Layer

- ESs must be packetized for transport in most scenarios. MPEG-4 defines a layer contains time stamps—the *sync layer* (SL)
- SL packet consists of a header and the payload.
  - Header is configurable, done through SL configuration descriptor
- SL packetized streams are transported using an underlying transport protocol

34

Distributed Content, Time Bases, and OCR Streams

- The speed of time (or time base) of two content authors may be off by just a small amount, which causes a drift between them during long run



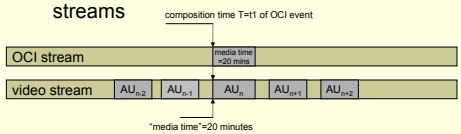
drift between different time bases

- *object clock reference* (OCR) time stamps are sent on a regular basis and contain the sender's current time base.

35

Media Time

- Unicast or local playback: start time of stream corresponds to media time zero is sufficient
- Broadcast application: random tune-in capability?
  - Media time descriptor in the OCI
  - Independently edit media time from media streams



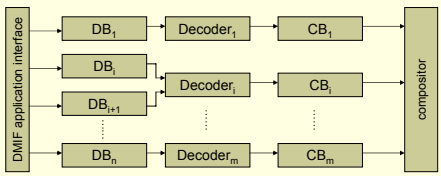
36

System Decoder Model

- The model of MPEG-4 terminal is composed of several elements
  - Profile and level mechanism
  - System Decoder Model (SDM)
    - Buffering
    - Timely processing streamed data

37

System Decoder Model (cont)



Buffers in the system decoder model

38

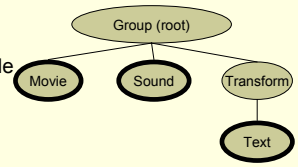
Outline

- Object Descriptions
- Semantic Description and Access Management
- Timing Model and Synchronization of Streams
- Introduction to Scene Description

39

Basics of BIFS

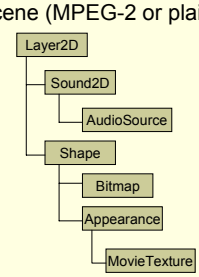
- MPEG-4 scene description—*binary format for scenes* (BIFS), the framework defines how objects are combined together
- Building scene as hierarchical structure or scene tree
  - Leaf node
  - Grouping node



40

Basic BIFS Features by Example

- Trivial Scene (MPEG-2 or plain DVD)



41

Basic BIFS Features by Example (cont)

- Movie with Subtitles
- Icons and Buttons
- Slides and Transitions
- Simple 3D scene

42