Multiplexing Digital Multimedia Data

Samuel Rivas
LambdaStream Servicios Interactivos, Spain

Miguel Barreiro
LambdaStream Servicios Interactivos, Spain

Víctor M. Gulías
University of A Coruña, MADS Group, Spain

INTRODUCTION

Multiplexing is the process of combining several independent signals to build another one from which it is possible to recover any of the original signals. This way, several sources of information can share one single transmission channel. The opposite operation is demultiplexing, or demuxing.

Multiplexing devices are called multiplexers, or muxers. Demultiplexer devices are called demultiplexers, or demuxers. A multiplexed stream is usually called multiplex.

For telecommunications, channel sharing usually aims for using one single physical channel to transmit several data streams. In multimedia, we assume an established digital data path, so a channel is a logical connection. The goal is to divide the complexity of digital multimedia systems in different access levels. For the example in Figure 1a, multiplex 1 hides the individual details of stream 1 and 2, and multiplex 2 adds a new access layer hiding the details of multiplex 1 and stream 3.

A typical application of multiplexing is mixing individual audio and video streams to build a new single stream for transmission or storage.

BACKGROUND

Streaming and Data Streams

Real-time multimedia data are usually transmitted using streaming techniques: data are streamed over a digital connection with a certain bitrate. Streamed data are called bitstream or stream.

Streaming is useful to transmit data representing digital signals that vary in time (e.g., video or audio). Clients interpret input streams at the time they receive it, as opposed to static data, where they wait until the transmission is complete.

Clients can start receiving a stream in the middle of a transmission. This enables digital broadcasting, where a server is transmitting a data stream continuously and clients "tune in" when they want (e.g., digital TV).

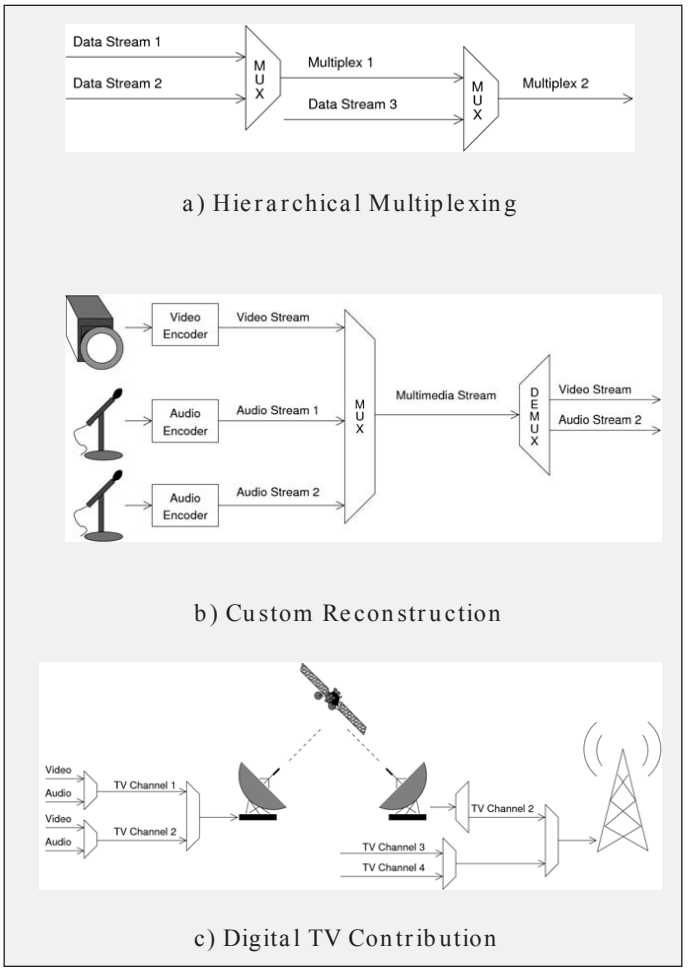
Elementary Streams and Multiplexed Streams

Multiplexing several streams results in a new stream. Already multiplexed streams may be further multiplexed with other streams to construct higher level streams. The leaves of any multiplexing hierarchy (data streams 1, 2, and 3 in Figure 1a) are called elementary streams. Elementary streams cannot be further demultiplexed.

Typical types of elementary streams are video, audio, subtitles, event streams, and so forth.

An advantage of this separation is that each elementary stream can be coded using specialised algorithms. Another benefit is that one single multimedia stream can carry different elementary streams for the same logical object, letting end users configure the final representation of the multimedia stream. For example, in Figure 1b, two audio streams are multiplexed in the multimedia stream. In the client side, the user selects one of them from the multiplex to reconstruct the media according with his preferences.

Figure 1. Multiplexing applications



Static Data

Apart from multiplexed real-time streams, multimedia streams contain also non real-time data. These data include metadata and static data. Metadata are information about the whole media stream and information about the multiplexed streams it contains. Static data may be of many types: teletext, electronic programming guide, interactive programs.

For storage, static data may be saved in chunks in any place of the media container. However, in streaming environments, specially in broadcasting, clients may start the reception at any point in the original stream. Thus, these data chunks must be transmitted periodically so that any client can receive them.

In Figure 2a, the grey block represents metadata that the demuxer needs to interpret the rest of the multimedia stream. In a random access storage, metadata may be saved anywhere because the demuxer can access it prior to start reading the multimedia stream. However, in broadcast environments, a demuxer that access the stream in t_1 cannot start demultiplexing it until t_2.

If a metadata block has a considerable length, it may not be practical to send the whole data block for each repetition. For that cases, MPEG defines the data carousel (ISO/IEC, 1998). In a data carousel, static data are sent in chunks regularly. When all chunks have been transmitted, the carousel starts again with the first chunk (Figure 2b).

Temporal Dependencies

There is one important difference between multimedia multiplexing and other types of multiplexing: data