

# Tecnologías Multimedia - Study Guide -

## Milestone 6: Compressing the audio data with zlib

Vicente González Ruiz - Depto Informática - UAL

October 7, 2020

# 1. Description

It's time to reduce bandwidth consumption. The `pack()` and the `unpack()` methods can compress and decompress, respectively, the chunks that are handled. To compress and decompress, we will use a free codec named **DEFLATE**, which is based on LZSS and Huffman Coding [1].

## 2. What you have to do?

1. Create a class named `Intercom_minimal_zlib`, that inherits from `Intercom_minimal`, in which the methods `pack()` and `unpack()` are overridden to compress and decompress the chunks. Use the Python's standard library `zlib`. Store this class in a module named `intercom_minimal_zlib.py`.
2. Create a class named `Intercom_inter_zlib`, that inherits from `Intercom_inter` and `Intercom_minimal_zlib`, to allow the compression of the inter-decorrelated subbands.
3. Notice that, in general, the compression ratio provided by DEFLATE is higher when the samples of the channels (in the case of `Intercom_minimal`) or the coefficients of the subbands (in the case of `Intercom_inter`) are not interleaved (all the samples/coefficients of the first channel/subband go first, and next, all the samples/coefficients of the second channel/subband are placed).

### 3. Timming

You should finish this milestone at most in two weeks.

## 4. Deliverables

The modules `intercom_minimal_zlib.py` and `intercom_inter_zlib.py`. Store them at the **root directory** of your `intercom`'s repo.

## 5. Resources

- [1] Nelson M. and Gailly J. *The Data Compression Book*. M&T Books, 1996.