# Design problem:

To generate supervised machine learning models to detect specific animal behaviours, we require a training dataset of labelled example behaviours. These are typically generated by filming an animal wearing a device and then time matching instances from the accelerometer ‘trace’ with the videos. There is no single way this is achieved, and each project brings with it specific problems that often require custom solutions. In this case, we face two main problems:

1. Camera timestamp and accelerometer timestamps are mismatched
2. Accelerometer timestamp ‘drifting’ over time (progressively more incorrect)

To overcome these, we need a system that allows us to load in a trace and a video. We align the first marker (normally a ‘clap’) which sets the absolute difference between trace and video times. We then set this as time 0 for both media formats. Then when we load in subsequent videos, their position is calculated based on the relative difference from the original video. There needs to be a fair bit of flexibility though as the drift over time means each alignment jump will be progressively but unpredictably worse.

# Earlier work:

The sync\_station\_v2 that I am using as my starting base was designed by Chris Clemente in 2024 (included in the folder). It is an updated iteration of a program originally conceived and designed 2017-2021, which can be found at the old git repo: <https://github.com/cclemente/Animal_accelerometry>. The limitation with this method is that, since the introduction of the ‘zoom’ functionality, the time-jumps seems to have been decoupled.

# Current Worklog:

* 22/10: I am going through the existing code to familiarise myself, update the formatting to structured functions as well as renaming all variables to clear and obvious names.
* I have split the original script into multiple smaller scripts where each script has a dedicated functionality. Each of these are then drawn together into the main script.
* While the figure app designer is useful for aesthetics of the app, it makes it hard to rapidly iterate design, and can easily decouple functionality. Therefore, I am going to be hardcoding the design in this version.