Atomic20

2 April 2015

Rapt project for Beneful/Purina

## What we’ve done:

We took the proof of concept and made a wrapper function that can be included in a web page as a separate javascript file.

When the client adds their iframe and the audio file, they can call the function name and then pass in five parameters:

1. The name of the video player iframe
2. The dom id of the audio file
3. The time offset (in seconds) for the start of the second audio track
4. The button label for audio track 1
5. The button label for audio track 2

Example:

dualAudio(myiframe, myaudio, 30, human, cheetah)

## What we’ve assumed

The brief specified accommodating for audio file load issues, buffering, and sync. After taking a close look we believe these issues are covered by the existing Rapt API calls.

The audio file is loaded using the same API call as loading the video files. We assume that the Rapt API already does a superb job at handing load issues, buffering, etc, and this code continues to leverage those capabilities and any future improvements.

The audio file doesn’t start playing until the video starts to play and the timecode of the audio tracks the timecode of the video. Provided the original file syncs well with the video, we see no reason why the sync shouldn’t be accurate in this use case as well.

## Questions

Do you need to support the use case where the client has two audio files rather than just one? Best practice is most likely to use one audio file, but…

The existing code calls a data.action function to toggle between the audio timecodes. Documentation on this call doesn’t appear to be in the included API docs, so we were unable to refine/alter this call.

This is relevant with regards to the labelling and styling of the audio toggle buttons. Presumably the display text of the buttons would be set by the label values that the client passes through to this new function, however, it doesn’t work that way currently.

With a bit more information, we can support some refinement of the button display possibilities, and we’re happy to do so.

# Bottom line

Let us know if what you need on this project as you move forward, and we’re happy to revisit and add functionality and/or additional safeguards.