The Problem with Timecode

Derek Schweickart
Senior Workflow Architect, Innovation
Frame.io

Rob Loughlin
Senior Innovation Technology Specialist
Frame.io

Timecode is NOT Time

The purpose of timecode

- The origin of timecode was as a solution for video tape editing, which was a difficult and imprecise process before ST-12 was proposed in 1970
- Timecode was meant to represent a physical location on a piece of media, that *referenced* time, but does not *represent* time.
- This has not changed with digital, where timecode is meant to locate a specific location in a group of digital bits, e.g. a single frame.
- The concept of “Reel” or “Tape” doesn’t apply well to digital files.
- Using timecode to sync video to audio is actually **not** the purpose of timecode.
- Timecode is **not** meant to identify a moment of “real time.”

TIMECODE IS NOT A CLOCK

The time is now

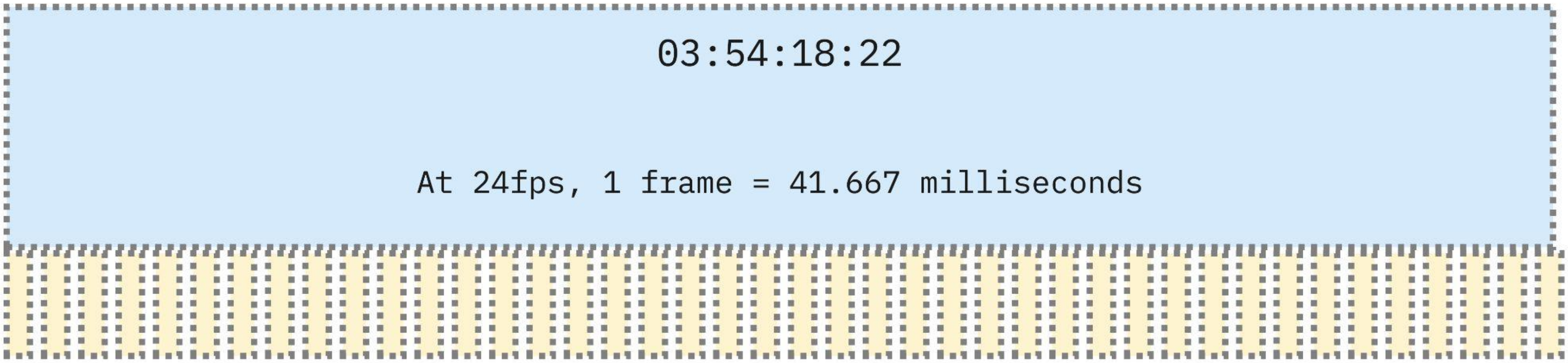
- Acquisition workflows have evolved into data-rich, file-based processes
- Improved global networking has enabled the Cloud to become a core element of audio and video workflows
- In the near future, moving audio and video as signals will be rare; instead they will be transmitted as data packets

We need a new time standard to further empower our data and cloud driven workflows

The problem of perspective

- TIMECODE IS NOT TIME OF DAY
- Using timecode as time of day was adopted mainly because of the value of syncing audio with video...two distinct perspectives.
- When using this method with multiple cameras, we end up with duplicate timecode values.
- Modern workflows involve even more perspectives and generate more data that needs to sync together (lens data, telemetry, haptic, motion capture, etc.)
- Timecode is NOT suited for this level of complexity.
- Even syncing audio and video together is an approximation

The problem of size



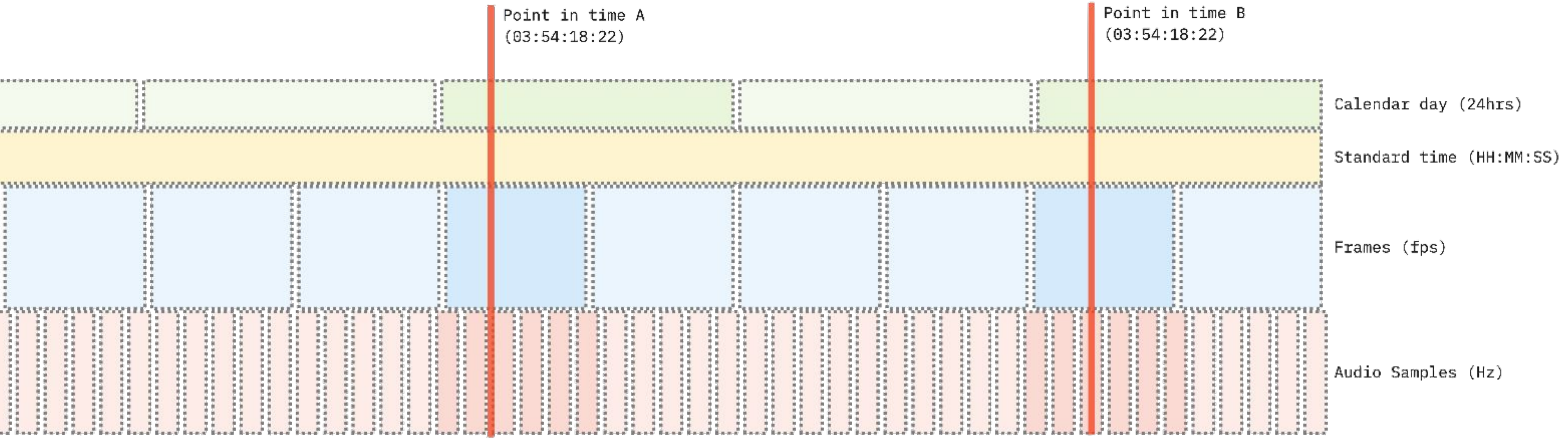
A single timecode value is a range of time

The problem of size

- Time based media is freezing a continuous range of time at a certain sample rate within certain boundaries (start and end)
- A timecode value doesn’t represent a specific moment in time, but rather a range of time (ie, 1/24th of a second)
- A framerate is an image sample rate, but audio is sampled at much higher rates than video
- Because audio has a higher resolution, video often needs to be synced at “sub-frame” intervals
- Timecode is limited to a 24 hour clock, creating even more duplicate values.

Timecode is both too broad and too narrow

The problem of size



Timecode is both too broad and too narrow

The problem of frequency

- The size of a time range can be variable.
- Timecode has a fixed framerate reference which does not allow for variable frame rates or mixed frame rates.
- Variable frame rates have uses for both decoding/exhibition and for creative effect
- Currently, there is no standardized way to track variable frame rates
- As an example, in overcranking workflows, the unit of time of acquisition is different than the unit of time of exhibition.

Timecode is both too broad and too narrow

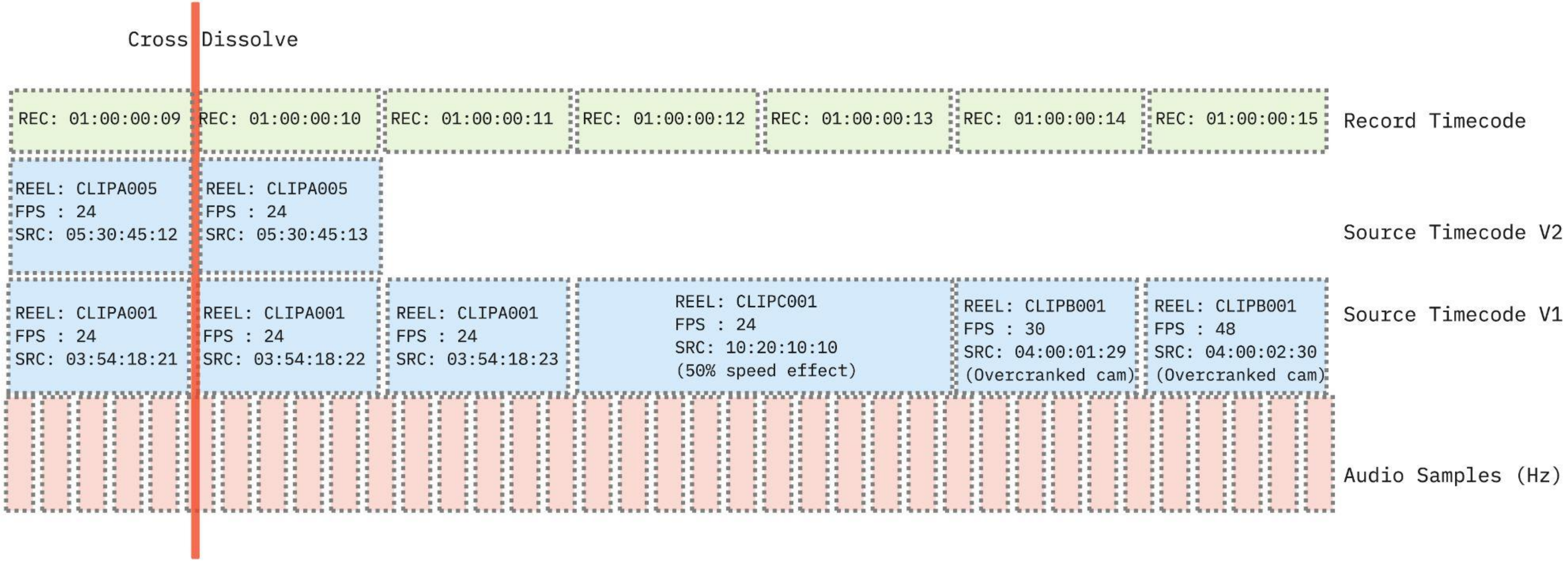
The problem of location

- In the context of cloud production, we can imagine that a video frame is created in one time, but a note given on that frame might be given in another timezone.
- When a file or video stream exists in the cloud, it doesn't really exist in any particular time zone because it can be accessed from anywhere with only a modest amount of latency.

The problem of editing

- For a given frame, it has a source identifier which is fixed, and an record identifier which is variable.
- This problem compounds as more assets are added and more generations are made.
- ST-12 cannot hold both source and record values at the same time.
- Besides duplicates, we have multiple time references that relate to the **same** individual frame of media.

Layers of Time



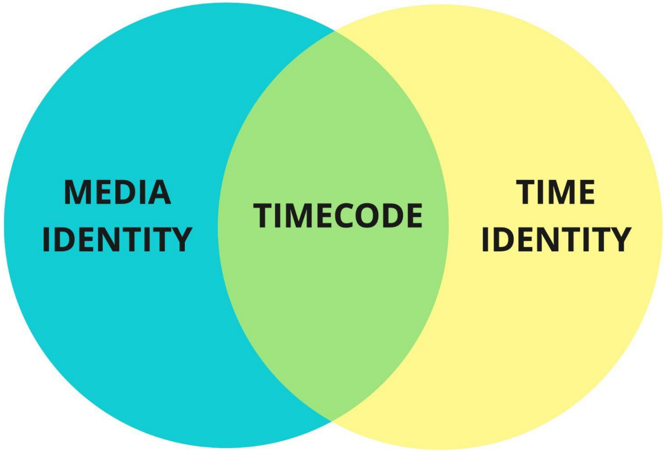
Timecode as Identification

- SMPTE ST-12 was a way to standardize how machines could **identify** an individual frame on a piece of video tape
- We now need to think about how to identify the smallest units of media within modern media technologies.
- Identity in the modern age is a more challenging topic than in the days of tape.
- This topic intersects with the rigorous demands on information security in an increasingly digital and distributed world.

Timecode as Identification

- Timecode is an overlap of identifying media and identifying real time.
- Because we use time-based media these notions get conflated, but they are distinct.

Media identity is
What



Time identity is
When

Media Identity Requirements

- Can locate a single unit of time-based media, e.g. video frame, audio sample
- Can carry a history of identity when media is used to create new media, e.g. in an edit.
- Can carry contextual data, allowing media to be related to other media, i.e. filename, production information, camera position, scene, take etc.
- Can provide ability to distinguish generations/proxies of media from their parents.

Time Identity Requirements

- Can give precision to at least a microsecond level
- Can sync to a world clock
- Contains date and timezone information
- Contains frame rate and frame duration information
- Can be human readable

TLX

- The digital birth certificate idea addresses media identity, and the High Precision Time Source addresses time identity.
- TLX offers much more in the way of information about the media, but doesn’t directly address multiple time labels or history.
- The notion of TLX Profiles shows the complexity of the issue in that the general functioning of TLX would vary depending on the application and use case.

Finding the Right Successor to Timecode

What do we need from a new standard?

- A new standard needs to identify the smallest unit of an asset as unique from other units and other assets
- A new standard should be able to identify how assets relate to each other in time
- A new standard should be data-rich
- A new standard should be able to be easily implemented into existing and future file containers
- A new standard should also have provisions for backwards compatibility with ST 12

What do we need from a new standard?

We started with the following assumptions. In the future, we assume that:

1. Video transmission will disappear, to be replaced by file transmission
2. All hardware will be connected to the internet
3. All original assets will transmit to the cloud on creation automatically
4. Video, audio, and time resolutions will increase for the purpose of super-sampling

Timecode + UTC

- SMPTE ST309 Date/Time in User Bits
- Pairing a timecode value with a UTC value attaches a unique time identifier
- UTC also allows for local offset, providing context for location
- UTC is a global standard and is retrieved from a common source
- UTC has nuances to work around and/or solve for, like leap seconds

Timecode + Frame Rate

- Because timecode is not a representation of time, it doesn't actually include information about frame rate. This data is traditionally managed elsewhere.
- By including frame rate and frame duration, we get more accurate information about the media which can be used both technically and creatively.

```
timecode = {
"utctime" : "2021-09-16T16:48:02Z",
"rate" : 24000/1001,
"frame_duration" : 1/48,
"utc_frame" : 3, #how many frames after the second boundary
}
```

Backwards compatibility

- It will be necessary to preserve backwards compatibility.
- One way to accomplish this with regards to creating an ST-12 compliant timecode string, would be to always carry the means to move between UTC and ST-12
- This could be achieved with an offset value.

```
timecode = {
"utc_time" : "2021-09-16T16:48:02Z",
"rate" : 24000/1001,
"frame_duration" : 1/48,
"utc_frame" : 3, #how many frames after the second boundary
"local_zone" : -5, #how to normalize the time to a local time
"utc_offset" : "2021-09-16T00:00:00Z", #offset back to 24hr clock
"tc_string" : "11:48:02:03" #time of day TC clock localized
}
```

Backwards compatibility

- Edited media will likely derive more value from arbitrary values like 01:00:00:00 that time of day
- The UTC offset could be a way to correlate world time to timeline timecode.

```
timecode = {
"utc_time" : "2021-09-16T20:48:02Z",
"rate" : 24000/1001,
"frame_duration" : 1/48,
"utc_frame" : 0,
"local_zone" : 0,
"utc_offset" : "2021-09-16T19:48:02Z",#note one hour offset
"tc_string" : "01:00:00:00"
}
```

Timecode layers

```
timecode = {
  [
    {
      "layer" : 0,
      "layer_name" : "source",
      "utc_time" : "2021-09-16T16:48:02Z",
      "rate" : 24000/1001,
      "frame_duration" : 1/48,
      "utc_frame" : 3, #how many frames after the second boundary
      "local_zone" : -5, #how to normalize the time to a local time
      "utc_offset" : "2021-09-16T00:00:00Z", #offset back to 24hr clock
      "tc_string" : "11:48:02:03" #time of day TC clock localized
     },
     {
      "layer" : 1,
      "layer_name" : "edit",
      "utc_time" : "2021-09-16T20:48:02Z",
      "rate" : 24000/1001,
      "frame_duration" : 1/48,
      "utc_frame" : 0,
      "local_zone" : 0,
      "utc_offset" : "2021-09-16T19:48:02Z",
      "tc_string" : "01:00:00:00"
     }
  ]
}
```

The ability to create layers or indexes of timecode objects would allow for the tracking of a frame as it moves through the edit phase.

Connecting layers is complex and requires further exploration.

Timecode context

```
timecode = {
"time_identity" : {
    "utc_time" : "2021-09-16T16:48:02Z",
    "rate" : 24000/1001,
    "frame_duration" : 1/48,
    "utc_frame" : 3, #how many frames after the second boundary
    "local_zone" : -5, #how to normalize the time to a local time
    "utc_offset" : "2021-09-16T00:00:00Z", #offset back to 24hr clock
    "tc_string" : "11:48:02:03" #time of day TC clock localized
                   },
"media_identity" : {
    "uuid" : "4ee3e548-3374-11ec-9f4f-acde48001122",
    "creator" : {
        "application" : "ARRI_ALEXA",
        "serial_number" : "818599130"
        },
    "user_data" : {
        "scene" : "101A",
        "take" : "01",
        "clipname" : "A001C001"
      }
}
```

Adding in context data can help with the media identity.

This is an area where the concept of a TLX profile is very appealing, as the requirements for defining media identity may vary across the industry.

Implementation

- Needs to be able to be encoded in existing file formats.
- Decoding/Parsing needs to be consistent.
- Standard should be extensible.
- Manufacturers should be able to extend the standard in predictable, parsable ways (like an ALE or EDL)
- A new standard designed this way can exist alongside implementations of ST 12 in the same file

Timecode Codec

- One thing to consider is that not all the data we are interested in changes per frame (camera, scene, take, etc), and yet we want access to it at each frame during playback.
- This seems like a very good case to intelligently compress the data in this model.
- Streaming Protocols
- Dolby Vision RPU (Reference Processing Unit) is an example of translating per frame metadata to an embedded stream.

Time Pipeline

- Video capture workflows have evolved to increase image resolution, color gamut, and file and data compression
- Color standards (like DolbyVision) and color workflows (like ACES) have expanded to create traceable pipelines of color data
- Object based audio workflows (like Dolby Atmos) have allowed more precision and more creativity in sculpting audio scapes
- A layered approach to recording time can allow an asset's time manipulation to be traceable and remain non-destructive
- A data-rich time label can help trace an asset's journey from delivery back to acquisition

Setting a Standard for the Future

- Image, audio, and ancillary data capture techniques have evolved to incorporate more context and allow for more advanced, automated workflows
- Post production workflows have responded and have been designed to handle much more data
- There have been relatively few advancements in the recording and capturing of time data for media workflows
- Time serves as the backbone to nearly all of our workflows and is a core definition of what we do

```
media = (image + sound) * time
```

- The way we record, transmit, translate, and interpret time needs to evolve alongside our video and audio workflows

It’s time to talk about time

Thank you

rloughlin@frame.io | dschweickart@frame.io