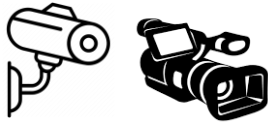


The purpose of this document is to define all the use cases of the backend service used to capture and share video

# 10 x 10 Design Goals

## 10 Active Sources



Dumb Streaming Sources



PTZ Controlled Streaming IP Camera Sources



PTZ Controlled Mobile Device



Manually Controlled Mobile Device

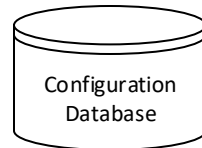


Local Camera Operator



Sensor Data Sources

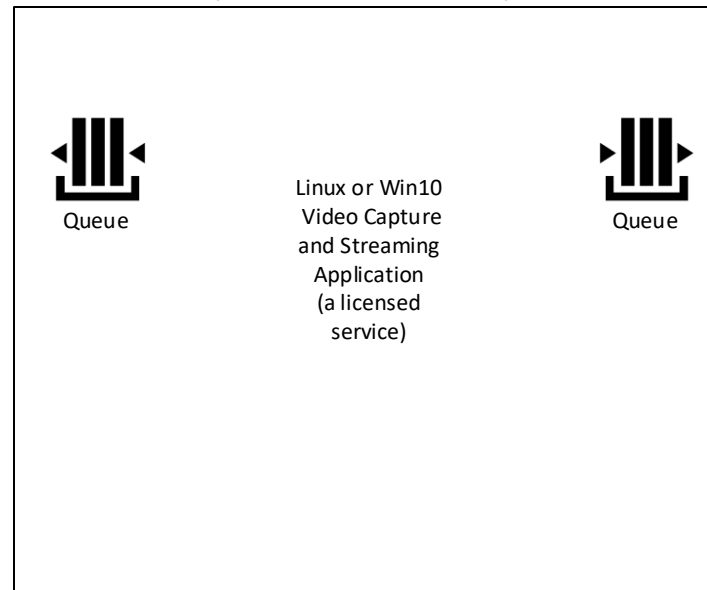
## Configuration



Configuration Database



Event Schedule Database



## 10 Active Consumers



Request Video File from Server



Request Video Stream from Server



Push Video to RTMP Streaming Networks

## Controllers



Tagging App



Mobile App



Event Schedule

## 10 x 10 Design Goals

### 10 Active Sources



Dumb Streaming Sources

The new video camcorders have recently added RTSP streaming capabilities over LAN or Wi-Fi. The main advantage of these devices is the high power zoom lens (10X) and better image quality than IP cameras.



PTZ Controlled Streaming Sources

The quality of PTZ IP RTSP cameras is getting better due to the use of these style cameras in video conferencing or remote education or corporate events. The only problem is high quality video with these devices is expensive



PTZ Controlled Mobile Device

Phone cameras are now more sophisticated with telephoto lenses, image processing, slow motion recording, low-light improvements, programmability and the new Wi-Fi 6 support. But, can we add a PTZ and remote control the phone?



Manual Controlled Mobile Device

The use of hand held device cameras by a coach or assistant will continue to be highest priority on small events like a golf or tennis lesson one-on-one between coach and student. Record the video and send to the server when complete.



Local Camera Operator

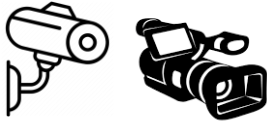
The big events like a football game will have a camera operator that controls the camera position and view, they will zoom in and out to show details and stop and start recording. This camera may need to be directly connected to the server.



Sensor Data Sources

There are a lot of sports already using digital sensors and data capture. We need to plan for this, but don't have a current requirement ready to implement.

10 Active Sources



Dumb Streaming Sources



PTZ Controlled Streaming Sources



PTZ Controlled Mobile Device



Manual Controlled Mobile Device



Local Camera Operator



Sensor Data Sources

***Ideas to Improve Performance  
and Scalability of the Capture***

- Control the camera resolution, frame rate and bit rate (quality). Not every camera needs the same resolution (i.e. 720P, 1080P, 4K).
- Disable audio track on recording if not needed.
- Operator can reduce the motion and panning of the camera. Motion generates larger files or poor quality when using variable BIT rate encoding.
- Not every device has the same priority. Use a method to control capture that ensures higher priority sources get the needed resources.
- Chunk video into smaller blocks if possible so that they can be restarted on playback if the network quality is poor or the signal is interrupted.
- Use encoding formats that are optimized for the requirements of the application.
- For bigger events; we may need to have more than one server in use.

## 10 x 10 Design Goals

### 10 Active Consumers

- Tablets and mobile devices are used to view video segments captured by the server .
- We need to be able to support up to 10 mobile devices during a live event.
- Use prioritization to ensure the users have control over performance in a constrained environment.
- The new Wi-Fi 6 will allow more concurrent sessions. May need different rules for 802.11ac and 802.11ax devices.
- The MQ design should improve reliable delivery. Optimize for reliability and security.
- Can we chunk the video to improve recovery?
- The tagging meta data is used to enable distribution.



Request Video File from Server

If the source video comes from an RTSP stream from a video camera to the server we will need to be able to route that video to a consumer device. How do we do that? What streaming technology is preferred? There are new streaming features coming from Apple and others.



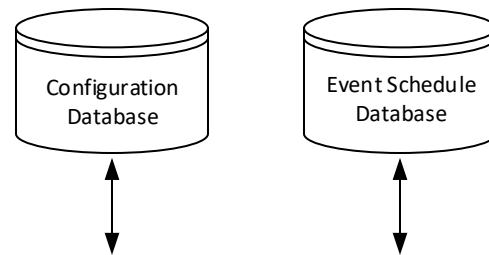
Request Video Stream from Server

The interest in streaming sports video to fans on the Internet is high, even for local club team with small numbers of fans. These services use RTMP for streaming rather than RTSP. What tools are available for conversion of compatability.



Push Video to RTMP  
Streaming Networks

### Configuration



We will need to provide configuration data to the server. The source of the data could be a local file on the Win10 service or in the Linux Strapi database or the cloud.

We need to be able to profile and customize the source and consumer types.

Need to be able to identify and authorize the users.

Should have groups of devices so multiple events could run concurrently.

Reserve devices at certain times for scheduled events.

The Admin configures the Editing app that receives the data.

The tagging app should be of two types, basic ODK type data and detailed data.  
Tagging data is used to control recording in real time.  
One tagging app should be able to control the whole network.  
The mobile app will be a future version of the Win10 editing app  
The calendar can also initiate and terminate recording unattended  
Might also add the Stream Deck if it is not too complicated (and closed architecture)

### Controllers



Tagging App



Mobile App



Event Schedule

### **Test Scenarios**

- Vary size, memory, configuration of server.
- How difficult will this be to implement on Strapi/Linux?
- Vary the number of sources and consumers to determine maximum users supported.
- Vary the resolution, encoder, bit rate.
- Vary the mix of devices.
- Vary the Wi-Fi standard from Wi-Fi 5 to Wi-Fi 6
- Different sports scenarios (football, volleyball, soccer, cricket, basketball)

### **Test PIP**

- Roster data
- Schedule data
- ODK data
- Detail tagging data
- Timeline synchronization data (timeline object with properties and methods)
- Video clips (mp4 or other)
- Logging data