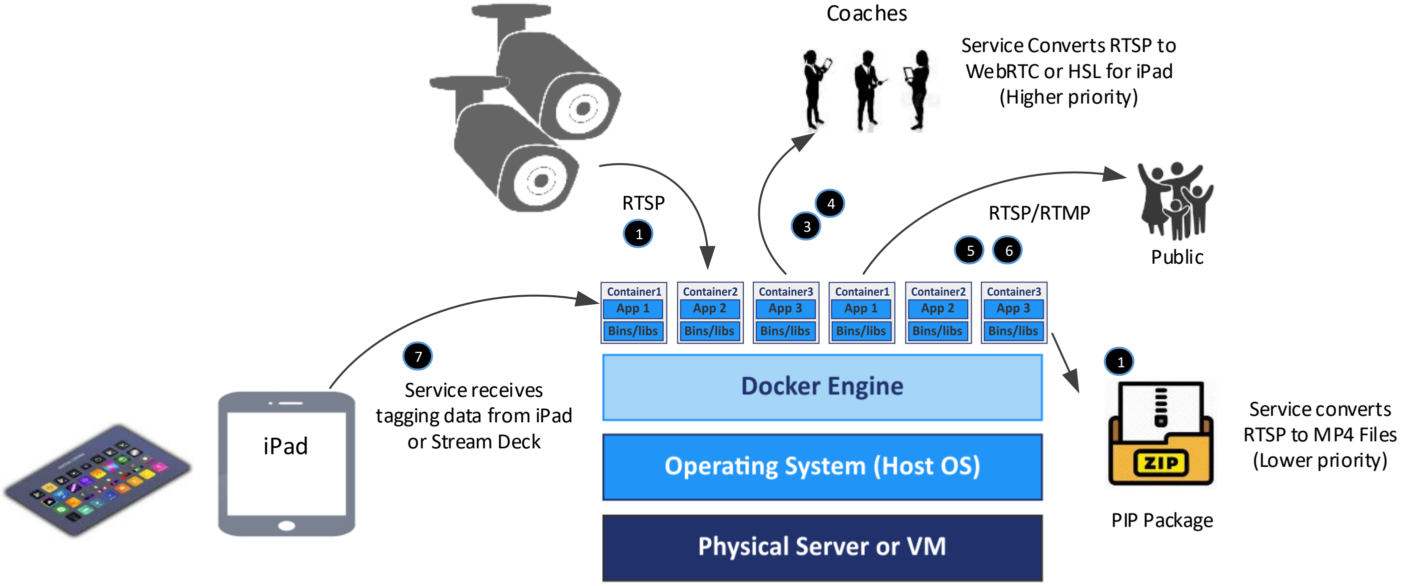
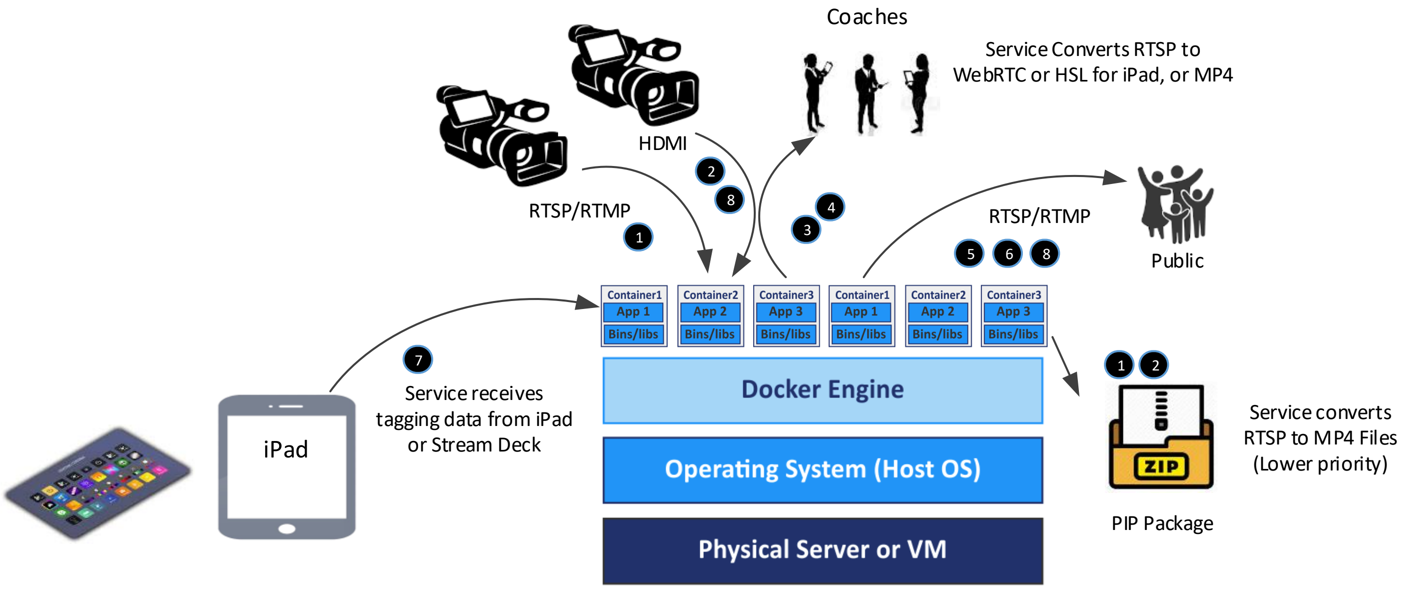
Scheduled Event Technical Requirements



Scenario:	Scheduled Event – Option 1
Source:	One or more IP Cameras
Connection:	RJ45 Ethernet on LAN to Linux Server
Video Codec:	H.264 or H.265
Resolutions	720P, 1080P, or 4K
FPS	Usually 30, some options 60 FPS; Bit Rate 2 – 10 Mbps per stream
Protocol:	RTSP (maybe RTMP from some newer cameras)
Recording:	Continuous play (this create one or two very long files in the PIP, the streams are viewed on demand)
Live Tagging:	Analyst using iPad, laptop or Stream Deck

Requirement 1:	Save RTSP streaming files in a PIP on server for analysis	Feature: Extract compressed video from RTSP, save as MP4 files on Linux Server
Requirement 2:	Save camera HDMI files in a PIP on server for analysis	Feature: Not possible with IP Cameras as video source
Requirement 3:	Coaches review plays, on demand, on iPad	Feature: Server converts RTSP stream to HSL or WebRTC, links with tagging data
Requirement 4:	Coaches review plays, on demand, on Win10 laptop	Feature: Server converts RTSP stream to HSL or WebRTC, links with tagging data
Requirement 5:	Stream video to 1 – 2 web browsers on Internet	Feature: Fast Internet connection, pass RTSP to small audience on web browsers
Requirement 6:	Stream video to dozens of fans or more, on the Internet	Feature: Convert RTSP stream to RTMP, send video stream to social media sites
Requirement 7:	Basic tagging using iPad or Stream Deck	Feature: Send tagging data live updates to server to provide time sync with video
Requirement 8:	Add graphics, score, and narration to stream	Feature: Not possible with IP Cameras as video source without special equipment

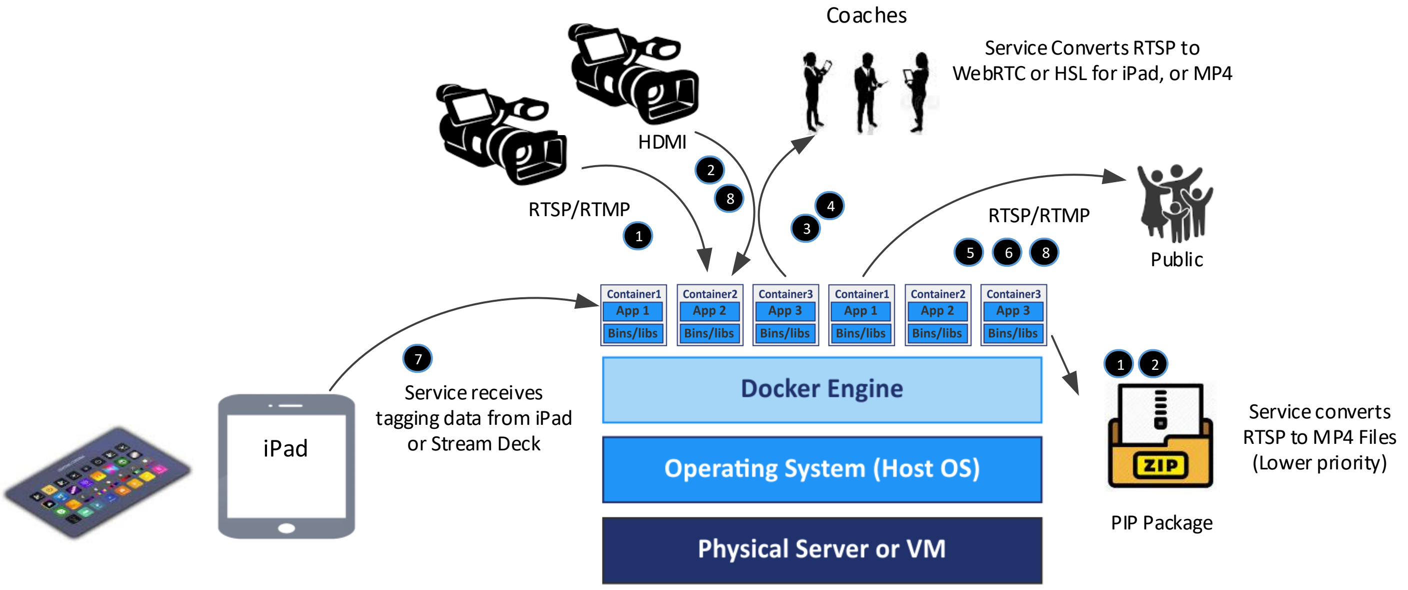
Concerns: TBD



Scenario: Scheduled event – option 2
Source: One or more camcorders with streaming feature
Connection: RJ45 Ethernet on LAN or Wi-Fi, and/or HDMI port capture to Linux Server
Video Codec: H.264 or H.265
Resolutions 720P, 1080P, (4K streaming not currently available from camcorders)
FPS 30, 60; Bit Rate 4 – 20 Mbps
Protocol: RTSP or RTMP, HDMI
Recording: Continuous play with occasional intermissions (this creates multiple longer files, no Internet streaming when not recording)
Live Tagging: Analyst using iPad, laptop or Stream Deck

Requirement 1:	Save RTSP streaming files in a PIP on server for analysis	Feature: Extract compressed video, save as MP4 files on Linux Server
Requirement 2:	Save camera HDMI files in a PIP on server for analysis	Feature: Capture card to convert HDMI to files; must copy files to iPad for review
Requirement 3:	Coaches review plays, on demand, on iPad	Feature: Server converts RTSP stream to HSL or WebRTC, links with tagging data
Requirement 4:	Coaches review plays, on demand, on Win10 laptop	Feature: Server converts RTSP streams to required format, may also move mp4 files
Requirement 5:	Stream video to 1 – 2 web browsers on Internet	Feature: Fast Internet connection, pass RTSP to small audience on web browsers
Requirement 6:	Stream video to dozens of fans or more, on the Internet	Feature: Convert RTSP stream to RTMP, send video stream to social media sites
Requirement 7:	Basic tagging using iPad or Stream Deck	Feature: Send tagging data live updates to server to provide time sync with video
Requirement 8:	Add graphics, score, and narration to stream	Feature: Possible with Camcorder over HDMI as video source

Concerns: Using both RSTP capture and HDMI video results in an incompatible mix of sources, some video can be streamed, some needs to be copied



Service Converts RTSP to WebRTC or HSL for iPad, or MP4

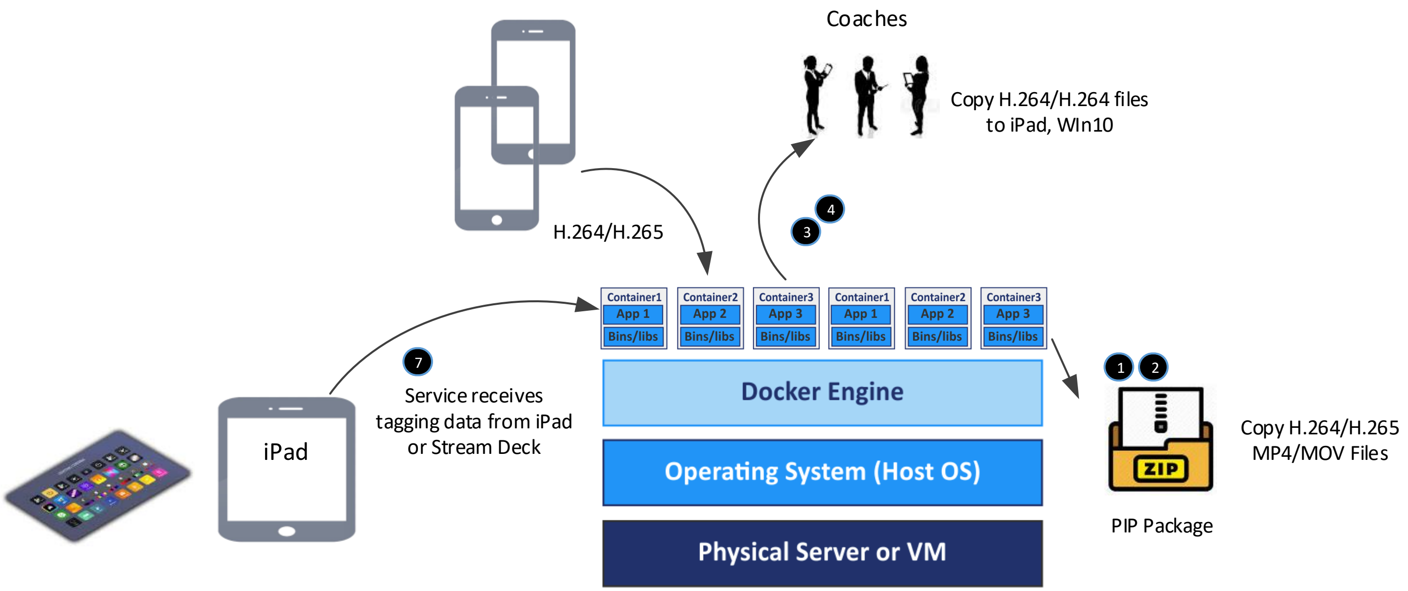
Service receives tagging data from iPad or Stream Deck

Service converts RTSP to MP4 Files (Lower priority)

Scenario:	Scheduled event – option 3
Source:	One or more camcorders with streaming feature
Connection:	RJ45 Ethernet on LAN or Wi-Fi, and/or HDMI port capture to Linux Server
Video Codec:	H.264 or H.265
Resolutions	720P, 1080P, (4K streaming not currently available from camcorders)
FPS	30, 60; Bit Rate 4 – 20 Mbps
Protocol:	RTSP or RTMP, HDMI
Recording:	Record every play, but *pause recording* between plays (this creates many smaller, shorter files, no Internet streaming when not recording)
Live Tagging:	Analyst using iPad, laptop or Stream Deck

Requirement 1:	Save RTSP streaming files in a PIP on server for analysis	Feature: Extract compressed video, save large number of MP4 files on Linux Server
		Feature: Merge the many small MP4 clips into one larger file for convenience
Requirement 2:	Save camera HDMI files in a PIP on server for analysis	Feature: Capture card to convert HDMI to files; must copy files to iPad for review
Requirement 3:	Coaches review plays, on demand, on iPad	Feature: Probably easiest to copy the smaller files to the iPad as needed
Requirement 4:	Coaches review plays, on demand, on Win10 laptop	Feature: Probably easiest to copy the smaller files to the laptop as needed
Requirement 5:	Stream video to 1 – 2 web browsers on Internet	Feature: Not very useful if only short segments are created, consider options
Requirement 6:	Stream video to dozens of fans or more, on the Internet	Feature: Not very useful if only short segments are created, consider options
Requirement 7:	Basic tagging using iPad or Stream Deck	Feature: Send tagging data live updates to server to provide time sync with video
Requirement 8:	Add graphics, score, and narration to stream	Feature: Not very useful if only short segments are created, consider options

Concerns: If only short 10 – 20 second clips are available, may decide to only send highlight scenes to social media

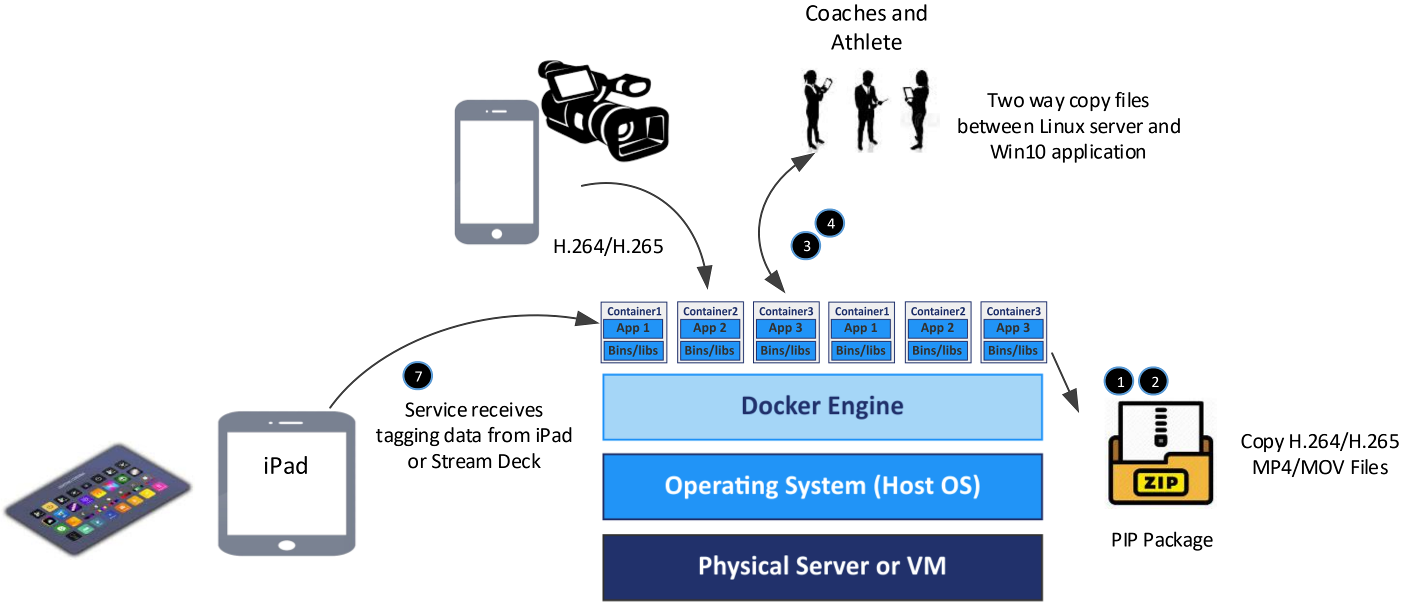


Scenario: Scheduled event – option 4
Source: Multiple users with mobile devices such as iPad or Android phones
Connection: Using Wi-Fi to send many short MP4 or MOV clips to the Linux Server
Video Codec: H.264 or H.265
Resolutions 720P, 1080P, 4K
FPS 30, 60; Bit Rate 5 – 10 Mbps
Protocol: MP4 or MOV
Recording: Record every play, but *pause recording* between plays (this creates many smaller, shorter files, no Internet streaming when not recording)
Live Tagging: Analyst using iPad, laptop or Stream Deck

Requirement 1: Save MP4 video files in a PIP on server for analysis — Feature: May merge many small MP4 clips into one larger file for convenience
Requirement 2: Save camera HDMI files in a PIP on server for analysis — Feature: Not available on mobile devices
Requirement 3: Coaches review plays, on demand, on iPad — Feature: Copy the smaller files to the iPad as needed
Requirement 4: Coaches review plays, on demand, on Win10 laptop — Feature: Copy the smaller files to the laptop as needed
Requirement 5: Stream video to 1 – 2 web browsers on Internet — Feature: Not practical for more than 1 viewer from a phone
Requirement 6: Stream video to dozens of fans or more, on the Internet — Feature: Not possible with MP4 files unless streaming app is available
Requirement 7: Basic tagging using iPad or Stream Deck — Feature: Send tagging data live updates to server to provide time sync with video
Requirement 8: Add graphics, score, and narration to stream — Feature: Not available on mobile devices

Concerns: This is a workable solution for smaller events, but with lower quality video recording from small lenses

Requirements Summary	**Status**
Convert H.264 RTSP streaming video to an .mp4 file.	Currently working
Convert H.265 RTSP streaming video to an .mp4 file.	May require some changes to support H.265
Convert RTSP streaming video to RTMP protocol in Linux server.	Some open source tools exist
Convert RTSP streaming video to WebRTC protocol in Linux server.	Some open source tools exist
Convert RTSP streaming video to HSL protocol in Linux server.	Some open source tools exist
Host a streaming server in a Docker container.	Some open source tools exist
Stream WebRTC or HSL from Linux server to iPad or Win10 laptop.	Some open source tools may exist
Copy mp4 video files from Linux server to iPad or Win10 laptop.	Some open source services exist
Stream RTSP video from IP camera to open Internet.	Not really practical for security and performance reasons
Stream RTSP video from Linux server to open Internet.	Still only supports a couple viewers without performance impact
Stream RTMP video to social media (i.e. YouTube).	Some open source tools exist, one to many streaming
Merge small mp4 files into larger clips for handling.	Some open source tools exist
Control IP cameras from Stream Deck or iPad.	Evaluate potential for device control
Control camcorder from Stream Deck or iPad.	Evaluate potential for device control
Control Linux Server workflow from Stream Deck or iPad.	Evaluate potential for device control
Linux server needs to perform multiple tasks concurrently.	Will be part of the microservices design



Two way copy files between Linux server and Win10 application

Service receives tagging data from iPad or Stream Deck

Copy H.264/H.265 MP4/MOV Files

Scenario:	Unscheduled event – option 5
Source:	DSLR Camera or MicroSD card; iPhone or Android
Connection:	Use laptop or desktop to copy video clips to Pip
Video Codec:	H.264 or H.265
Resolutions	720P, 1080P, 4K
FPS	30, 60, 120, 240; Bit Rate 10 – 40 Mbps
Protocol:	MP4 or MOV
Recording:	Record scenes as needed
Post Tagging:	Tagging after the event or scene was recorded

Requirement 1:	Save MP4 video files in a PIP on server for analysis	Feature: May merge many small MP4 clips into one larger file for convenience
Requirement 2:	Save camera HDMI files in a PIP on server for analysis	Feature: Not available on mobile devices
Requirement 3:	Coaches review plays, on demand, on iPad	Feature: Not applicable
Requirement 4:	Coaches review plays, on demand, on Win10 laptop	Feature: Not applicable
Requirement 5:	Stream video to 1 – 2 web browsers on Internet	Feature: Not applicable
Requirement 6:	Stream video to dozens of fans or more, on the Internet	Feature: Not applicable
Requirement 7:	Basic tagging using iPad or Stream Deck	Feature: Perform detailed tagging during analysis stage in Win10 application
Requirement 7:	Add graphics, score, and narration to video	Feature: Use tagging feature in Win10 application

Concerns: This is my golf example. Need to elaborate on the tagging and export video highlights to the cloud.