

Использование RTCMTLVideoView для отображения видео

В сборке 2.6.6 добавлена возможность использования `RTCMTLVideoView` для отображения видео, если устройство поддерживает данный вид.

Разберем автоматическое переключение между видами для отображения локального видео на примере [TwoWayStreaming](#)

1. Определяем, поддерживается ли Metal View

[code](#)

```
id<MTLDevice> localDevice;
#ifdef __aarch64__
    localDevice = MTLCreateSystemDefaultDevice();
    ...
#endif
```

2. Если `MTLCreateSystemDefaultDevice()` возвращает не `nil`, создаем объект `RTCMTLVideoView`

[code](#)

```
...
#ifdef __aarch64__
    localDevice = MTLCreateSystemDefaultDevice();
    if (localDevice) {
        RTCMTLVideoView *localView = [[RTCMTLVideoView alloc] init];
        localView.delegate = self;
        _localDisplay = localView;
    }
#endif
...
```

3. Если `MTLCreateSystemDefaultDevice()` возвращает `nil`, создаем объект `RTCEAGLVideoView` [code](#)

```
...
if (!localDevice) {
    RTCEAGLVideoView *localView = [[RTCEAGLVideoView alloc] init];
    localView.delegate = self;
    _localDisplay = localView;
}
...
```

Полный код примера

```
id<MTLDevice> localDevice;
#ifdef __aarch64__
    localDevice = MTLCreateSystemDefaultDevice();
    if (localDevice) {
        RTCMTLVideoView *localView = [[RTCMTLVideoView alloc] init];
        localView.delegate = self;
        _localDisplay = localView;
    }
#elseif
    if (!localDevice) {
        RTCEAGLVideoView *localView = [[RTCEAGLVideoView alloc] init];
        localView.delegate = self;
        _localDisplay = localView;
    }
    _localDisplay.translatesAutoresizingMaskIntoConstraints = NO;
```