

# Управление подложкой при отображении потока в Android

## Изменение цвета и прозрачности подложки

По умолчанию, подложка для отображения видеопотока `SurfaceViewRenderer` имеет черный цвет. При необходимости, подложка может быть сделана прозрачной. Для этого нужно импортировать следующие модули:

```
import android.graphics.PixelFormat;
```

и воспользоваться методами `setZOrderOnTop()` и `setFormat()`:

```
localRender = (SurfaceViewRenderer) findViewById(R.id.local_video_view);  
localRender.setZOrderOnTop(true);  
localRender.getHolder().setFormat(PixelFormat.TRANSPARENT);
```

Для того, чтобы задать цвет подложки, необходимо вызвать метод `setBackgroundColor()`:

```
remoteRender = (SurfaceViewRenderer) findViewById(R.id.remote_video_view);  
remoteRender.setZOrderOnTop(true);  
remoteRender.getHolder().setFormat(PixelFormat.TRANSLUCENT);  
remoteRender.setBackgroundColor(0xFFFF8800); //orange in RGB
```

## Очистка последнего кадра

По окончании воспроизведения видео, на подложке остается последний кадр. При необходимости, кадр может быть очищен, при этом подложка будет залита черным цветом. Для этого нужно импортировать следующие модули:

```
import android.graphics.Canvas;  
import android.graphics.Color;  
import android.graphics.PixelFormat;  
import android.graphics.PorterDuff;
```

и вызвать метод `release()`:

```
playStream.stop();  
playStream = null;  
remoteRender.release();
```

После вызова данного метода никакие операции с подложкой невозможны.

Если этот способ не работает, или подложка должна быть использована повторно, `SurfaceViewRenderer` может быть скрыт. Например, при нажатии кнопки `Play` делаем `remoteRender` видимым

```
mPlayButton.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View view) {
        mPlayButton.setEnabled(false);
        if (mPlayButton.getTag() == null ||
Integer.valueOf(R.string.action_play).equals(mPlayButton.getTag())) {
            remoteRender.setVisibility(View.VISIBLE);
            ...
            playStream = session.createStream(streamOptions);
            ...
        }
    }
});
```

Затем, когда нажата кнопка `Stop`, останавливаем воспроизведение и делаем `remoteRender` невидимым

```
...
    } else {
        playStream.stop();
        playStream = null;
        remoteRender.setVisibility(View.GONE);
    }
    ...
}
});
```