

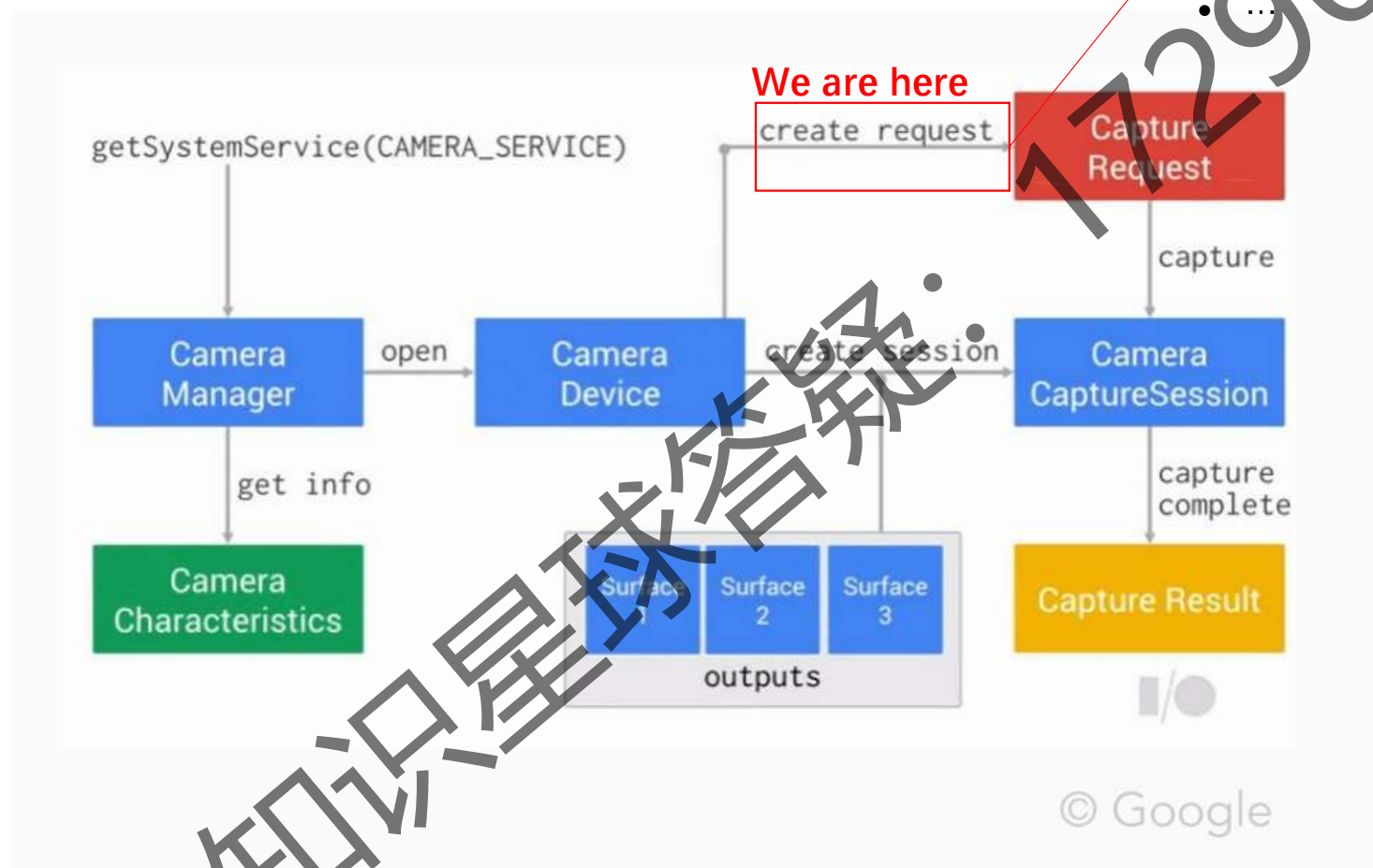
Android Camera2 API专题

第23讲

CaptureRequest详解

课程体系

- **CaptureRequest**
- 操作VendorTag
- 获取Physical Camera流
- ...



Agenda

- CaptureRequest介绍
- CaptureRequest.Builder介绍
- CaptureRequest Template type介绍
- 创建CaptureRequest.Builder
- 如何设置Physical Camera的CaptureRequest

```
java.lang.Object
└─ android.hardware.camera2.CameraMetadata<android.hardware.camera2.CaptureRequest.Key<?>>
    └─ android.hardware.camera2.CaptureRequest
```

CaptureRequest介绍

- CaptureRequest可以理解为告诉Camera底层如何处理某一帧数据，包括两部分内容
 - Settings
 - 硬件相关的参数：sensor, lens, flash等
 - ISP Processing Pipeline相关参数
 - 3A相关参数
 - Output Surfaces
 - CaptureRequest中带的Surfaces必须在创建Session时使用的Surface List的子集
- 通过CaptureRequest.Builder来创建CaptureRequest
- 通过不同的方式送给Camera底层
 - capture
 - captureBurst
 - setRepeatingRequest
 - setRepeatingBurst
- CaptureRequest类型（通过CaptureRequest#isReprocess()可以判断）
 - Regular: CameraDevice#createCaptureRequest
 - Reprocess: CameraDevice#createReprocessCaptureRequest

CaptureRequest.Builder介绍

- Builder模式：用于设置CaptureRequest参数和创建CaptureRequest对象

API	Description
set(Key<T> key, T value)	设置指定Key对应的Value值
setPhysicalCameraKey(Key<T> key, T value, String physicalCameraId)	设置指定Physical Camera中Key对应的Value值
addTarget(Surface outputTarget)	增加一路输出流
setTag(Object tag)	给CaptureRequest设置一个TAG，用于标识该CaptureRequest，底层在onCaptureCompleted回调时，可以通过Capturerequest#getTag()获取到
build()	创建CaptureRequest对象
get(Key<T> key)	获取指定Key对应的Value
getPhysicalCameraKey(Key<T> key, String physicalCameraId)	获取指定Physical Camera中Key对应的Value值
removeTarget(Surface outputTarget)	删除一路输出流

CaptureRequest Template type介绍

Template	Description
TEMPLATE_PREVIEW	创建适合用于 预览 的CaptureRequest，意味着：帧率不能向图像Quality妥协，所有Camera都保证支持
TEMPLATE_RECORD	创建适合用于 录像 的CaptureRequest，意味着：要出稳定的帧率，图像处理适合录像，除DEPTH_OUTPUT外所有的Camera都保证支持
TEMPLATE_STILL_CAPTURE	创建适合用于 拍照 的CaptureRequest，意味着：图像Quality不能向帧率妥协，除DEPTH_OUTPUT外所有的Camera都保证支持
TEMPLATE_VIDEO_SNAPSHOT	创建适合在 录像过程中用于拍照 的CaptureRequest，意味着：可以在不影响录像稳定帧率的情况下，最大化拍照的这张图像质量。除Legacy和DEPTH_OUTPUT外所有的Camera都保证支持
TEMPLATE_ZERO_SHUTTER_LAG	创建适合用于 App ZSL功能 的CaptureRequest，意味着图像Quality不能向帧率妥协，AE/AWB/AF处于AUTO模式，在支持PRIVATE_REPROCESSING 和 YUV_REPROCESSING的设备上保证能支持
TEMPLATE_MANUAL	创建适合用于 专业模式 使用的CaptureRequest，AE/AWB/AF都被Disable了，App可以手动控制3A参数，在支持MANUAL_SENSOR的设备上保证能支持

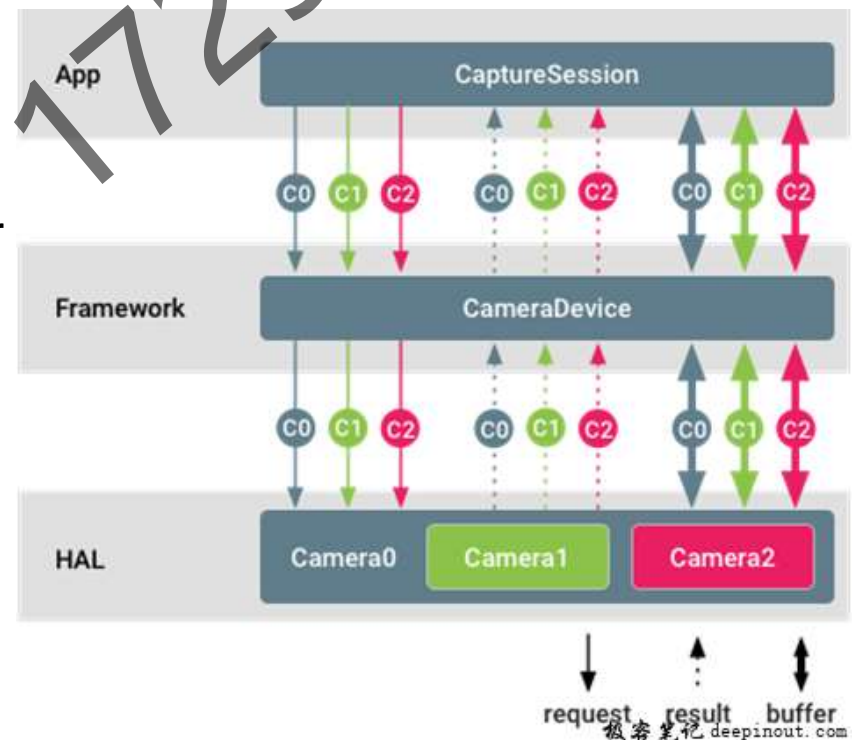
- 实验对比各种Template的参数值差异

创建CaptureRequest.Builder

API	Description
<code>createCaptureRequest(int templateType)</code>	根据Template Type创建CaptureRequest.Builder，注意：这里返回的是带一包默认值的，不同的CameraDevice，这包默认值会不一样，因此不能混用。
<code>createCaptureRequest(int templateType, Set<String> physicalCameraIdSet)</code>	根据Template Type和Physical CameraID Set创建CaptureRequest.Builder，通过Builder#setPhysicalCameraKey可以设置指定Physical Camera的参数。Physical Camera的参数只适用于有Physical Camera OutputConfiguration的情况（OutputConfiguration#setPhysicalCameraId）。physicalCameraIdSet来自CameraCharacteristics#getPhysicalCameraIds()
<code>createReprocessCaptureRequest(TotalCaptureResult inputResult)</code>	根据TotalCaptureResult创建用于Reprocess的CaptureRequest.Builder

如何设置Physical Camera的CaptureRequest

1. 判断能否单独控制Logical MultiCamera中的Physical Camera
(CameraCharacteristics#**getPhysicalCameraIds**)
2. 或者可以设置的Physical Camera Keys
(CameraCharacteristics#**getAvailablePhysicalCameraRequestKeys**)
3. 创建Physical Camera OutputConfiguration
(OutputConfiguration#**setPhysicalCameraId**)
4. 创建CaptureRequest.Builder
5. setPhysicalCameraKey
6. addTarget
7. 获取Physical TotalCaptureResult
(TotalCaptureResult#**getPhysicalCameraTotalResults**)



Thanks