

GetScreenshot和GetScreenshot2屏幕截图

当前屏幕截图有两种方法. 一个是轮询设备是否截图完成, 第二个是异步等待设备回复的截图. 推荐使用第二个

GetScreenshot轮询操作截图

操作流程:

1. 使用GetScreenshot通知下位机开始截图
2. 使用GetScreenshotStatus获取图片数据是否存在
3. 使用GetScreenshotData获取图片数据

```
<!--
1. 清理截图数据，并通知下位机开始截取一张截图
2. ##width取值范围{-int, int}，默认(小于等于0则自适应，大于帧缓存范围是则自适应)
3. ##height取值范围{-int, int}，默认(小于等于0则自适应，大于帧缓存范围是则自适应)
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <in method="GetScreenshot">
    <image width="##width" height="##height"/>
  </in>
</sdk>

<!-- 参数应答
1. ##result kCreateFileFailed 则启动截图失败，一般是节目未播放
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <out method="GetScreenshot" result="kSuccess">
  </out>
</sdk>
```

获取下位机截图状态, 用于辅助GetScreenshot

```
<!--
1. 获取下位机截图状态, 判断下位机是否存在截图数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <in method="GetScreenshotStatus">
  </in>
</sdk>

<!-- 参数应答
1. ##result 是kProcessing则没有图片数据, 如发了GetScreenshot则正在截图中, kSuccess
则存在截图数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <out method="GetScreenshotStatus" result="kSuccess">
  </out>
</sdk>
```

获取下位机截图数据, 用于辅助GetScreenshot

```
<!--
1. 获取下位机截图数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <in method="GetScreenshotData">
  </in>
</sdk>

<!-- 参数应答
1. ##data数据是图片转换为base64的数据, 还原后的数据内容是jpg图片数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <out method="GetScreenshotData" result="kSuccess">
    <image data="##data"/>
  </out>
</sdk>
```

GetScreenshot2异步操作截图

GetScreenshot2是异步截图的, 当你请求后, 你就可以继续做其他事情, 它不会阻塞设备操作.

同时这和GetScreenshot使用的是不同的缓存, 也就是说你不能使用上面的去检查设备是否有截图数据, 也没必要这样操作

在设备完成截图操作后, 会自动向你反馈数据, 而不需要再次请求一次. 当然, 这反馈速度是回复时间和截取屏幕所耗费的时间的总和的.

```
<!--
  1. 清理截图数据, 并进行下位机截取一张截图, 截图完毕后会进行回复数据
  2. ##width取值范围{-int, int}, 默认(小于等于0则自适应, 大于帧缓存范围是则自适应)
  3. ##height取值范围{-int, int}, 默认(小于等于0则自适应, 大于帧缓存范围是则自适应)
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <in method="GetScreenshot2">
    <image width="##width" height="##height"/>
  </in>
</sdk>

<!-- 参数应答
  1. ##data数据是图片转换为base64的数据, 还原后的数据内容是jpg图片数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="##GUID">
  <out method="GetScreenshot2" result="kSuccess">
    <image data="##data"/>
  </out>
</sdk>
```

ScreenShot屏幕节目(已弃用)

仅部分设备支持

```
<!--请求-->
<?xml version = "1.0" encoding = "utf-8" ?>
<sdk guid="##GUID">
  <in method="ScreenShot" />
</sdk>

<!--反馈
##image base64的图片数据
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="624aedc8936411b50636e01248789d98">
  <out method="ScreenShot" result="kSuccess">
    <screenshot image="##image"/>
  </out>
</sdk>
```

