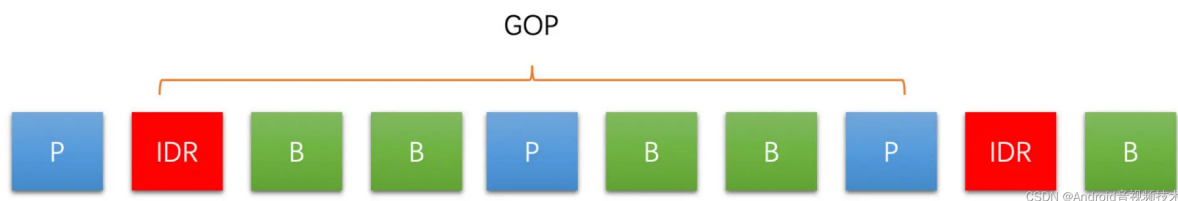


在上述所讲的B帧和P帧，它们是会去参考其他的帧来进行解析的。假如前面有一帧I帧出错了，那么后面参考它的P帧也出错了，接着参考这个I帧或P帧的其他P帧和B帧，也都相继出错。错误可能会不断传递下去。为了防止这种情况，H264 编码标准中规定，IDR 帧之后的帧不能再参考 IDR 帧之前的帧。也就是说我们在编码的时候可以编码一个IDR帧，这样这个帧之前的若有错误的帧，对这个IDR帧之后的数据的解析就不会有影响了。

实际上，在真实场景中，我们大概率会用IDR帧，而比较少用普通的I帧。不过如果你只用普通的I帧，标准也是支持的，但是不推荐。

1.3 GOP

GOP是代表从一个 IDR 帧开始到下一个 IDR 帧的前一帧为止的中间包含的所有IDR 帧、普通 I 帧、P 帧和 B 帧，也就是一个图像组。**GOP的大小就是两个IDR帧之间的间隔，也就关键帧间隔。**



GOP 越大，编码的 I 帧就会越少。相比而言，P 帧、B 帧的压缩率更高，因此整个视频的编码效率就会越高。但是 GOP 太大，也会导致 IDR 帧距离太大，点播场景时进行视频的 seek 操作就会不方便。并且，在 RTC 和直播场景中，可能会因为网络原因导致丢包而引起接收端的丢帧，大的 GOP 最终可能导致参考帧丢失而出现解码错误。

因此GOP不是越大越好，也不是越小越好，要根据实际场景来决定。

1.4 DTS & PTS

视频和音频的同步过程是一个你等我赶的过程，快了则等待，慢了就加快速度。这就需要有一个量来判断（和选择基准比较），到底是播放的快了还是慢了，或者正以同步的速度播放。在视音频流中的包中都含有DTS和PTS，就是这样的量。

DTS(Decoding Time Stamp)：解码时间戳；告诉解码器packet的解码顺序，也就是先解哪个包，才能解哪个包，需要按照顺序来，才能正确地一个一个解析出来

PTS(Presentation Time Stamp)：显示时间戳；指得是packet解析出来后，要显示的时候是想显示哪个再显示哪个，指明的数据的显示播放的顺序

在这里可能有人会有个疑问，这两个东西为什么会不同？一个一个解析，然后一个一个显示，这样的话不就两个应该是一样的吗？

在这里可能有人会有个疑问，这两个东西为什么会不同？一个一个解析，然后一个一个显示，这样的话不就两个应该是一样的吗？

其实，之所以两者会不同，主要也是跟上面提到的I、B、P帧有关，尤其是B帧，因为B帧要解析需要参考前面的或者后面的帧率，那么就代表着B帧没法按照显示顺序来解析，可能需要先解析后面的某个I帧，P帧，才能解析当前的这个B帧。这也是导致两者时间顺序不同的主要原因。

例如，假如一个视频序列，要这样显示I B B P，但是需要在B帧之前得到P帧的信息，因此帧可能以这样的顺序来存储I P B B，这样其解码顺序和显示的顺序就不同了，这也是DTS和PTS同时存在的原因。DTS指示解码顺序，PTS指示显示顺序。所以流中可以是这样的：

```
Stream : I P B B
```

```
DTS      1 2 3 4
```

```
PTS      1 4 2 3
```

CSDN @Android音视频技术