How to make IDR cuts — *instantaneous decoding refresh (IDR) cuts are smooth cuts that make for smooth splices.*

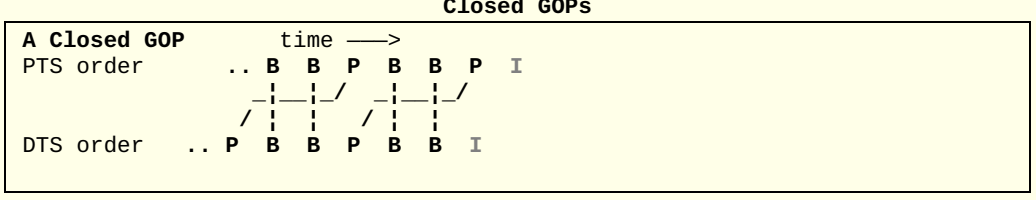
The OFFmpeg code shown applies solely to MPEG video found on DVDs & Blu-rays. Their video, audio, and subtitle streams have the same ticks-per-second (TPS), aka "time base". It doesn't apply to other transports such as MP4, MKV, etc. that typically have differing TPSes. The code could be altered to support separate TPSes but that's a little more complicated and this presentation is meant to be simple.

The OFFmpeg code shown is suitable only for constant frame rate (CFR) video. DVDs & Blu-rays are CFR.

Key: OFFmeg

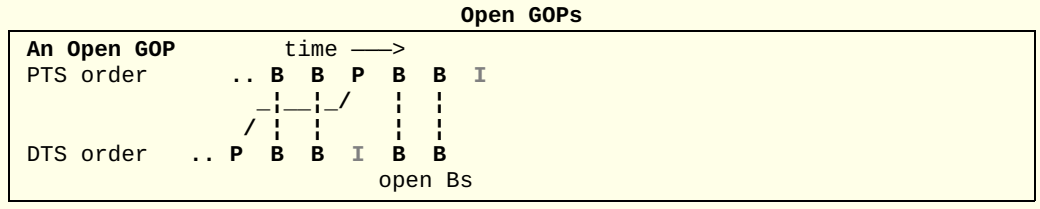
Closed GOPs

A Closed GOP



Open GOPs

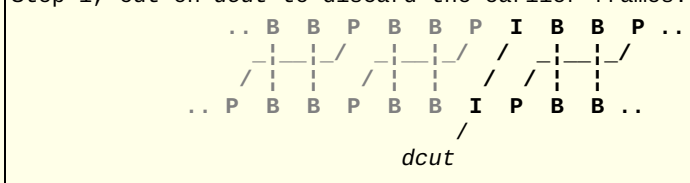
An Open GOP



A group of pictures (GOP) is an I-frame followed by P- & B-frames. The GOPs are shown as though they have PTS & DTS streams, but that's just to make time and physical frame order easier to visualize. Each is actually one stream of course, and PTSes and DTSes are actually just numbers.

To discard a segment that ends on a closed GOP:

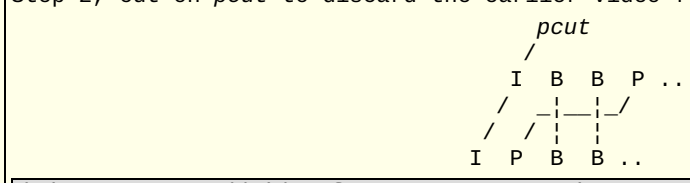
Step 1, cut on *dcut* to discard the earlier frames.



That there's anything important before dcut is unlikely, but if there is, set dstart to that DTS. Otherwise, set dstart to dcut.

```
// Set dstart = dcut to discard earlier non-video.
// Set dstart < dcut to preserve earlier non-video.
get'source'
(X(X            //all frames, one at a time
?dts'dstart..'( //match - is X at/past dstart?
:(X())        ) //n- drop X
))
{$v             //demux & focus on video stream
(X(X            //video frames, one at a time
?dts'dcut..'(   //match - is X at/past dcut?
:(X())      )   //n- drop X
))
```

Step 2, cut on *pcut* to discard the earlier video frames.



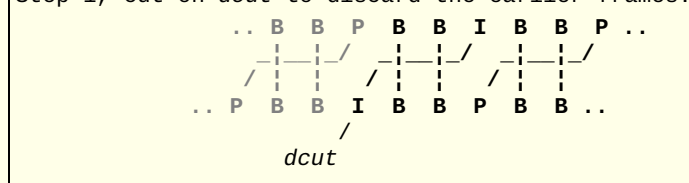

```
(X(X          //video frames, one at a time
?pts'pcut..'(  //match - is X at/past pcut?
:(X())      )  //n- drop X
))
$a$c$n$s}      //mux
put'target'end
```

See ***stride-match details***. *See* ***?-test details***.

The dts & pts & tpf properties are exact, but the dts & pts methods truncate when they write MPEG DTS & PTS tags.

To discard a segment that ends on an open GOP:

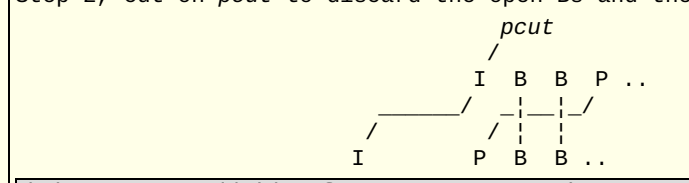
Step 1, cut on *dcut* to discard the earlier frames.



That there's anything important before dcut is unlikely, but if there is, set dstart to that DTS. Otherwise, set dstart to dcut.

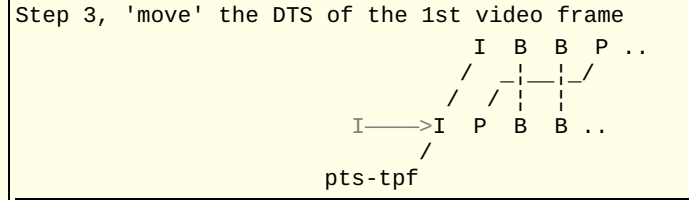
```
// Set dstart = dcut to discard earlier non-video.
// Set dstart < dcut to preserve earlier non-video.
get'source'
(X(X            //all frames, one at a time
?dts'dstart..'( //match - is X at/past dstart?
:(X())        ) //n- drop X
))
{$v             //demux & focus on video stream
(X(X            //video frames, one at a time
?dts'dcut..'(   //match - is X at/past dcut?
:(X())      )   //n- drop X
))
```

Step 2, cut on *pcut* to discard the open Bs and the earlier video frames.




```
(X(X          //video frames, one at a time
?pts'pcut..'( //match - is X at/past pcut?
:(X())      ) //n- drop X
))
```

Step 3, 'move' the DTS of the 1st video frame



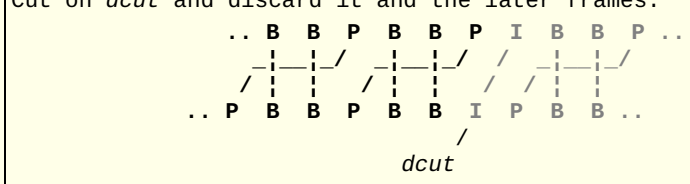

```
(X(X          //video frames, one at a time
dts'pts-tpf'  //new-match - rewrite X.dts
:X            //match
))
$a$c$n$s}     //mux
put'target'end
```

See ***stride-match details***. *See* ***?-test details***.

The dts & pts & tpf properties are exact, but the dts & pts methods truncate when they write MPEG DTS & PTS tags.

To discard a segment that follows a closed GOP:

Cut on *dcut* and discard it and the later frames.



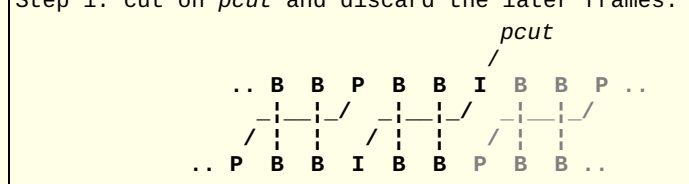

```
get'source'
(X(X          //match all frames, one at a time
?dts'dcut..'(  //is X at/past dcut?
(X())       )  //y- drop X
))
put'target'end
```

See ***stride-match details***. *See* ***?-test details***.

The dts & pts & tpf properties are exact, but the dts & pts methods truncate when they write MPEG DTS & PTS tags.

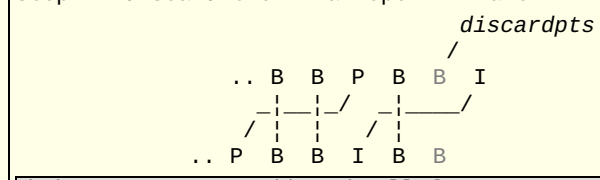
To discard a segment that follows an open GOP:

Step 1: cut on *pcut* and discard the later frames.



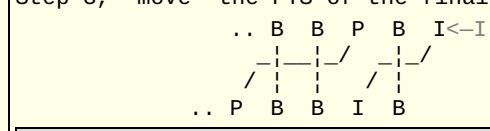

```
get'source'
(X(X          //match all frames, one at a time
?dts'..pcut'( //is X at/before pcut?
:(X())      ) //n- drop X
))
```

Step 2: discard the final open B-frame.




```
(X(X            //match all frames, one at a time
?pts'discardpts'( //is X at discardpts?
(X())           ) //y- drop X [note1]
))
```

Step 3, 'move' the PTS of the final video frame.




```
(X(X            //match all frames, one at a time
?dts'pcut'     ( //is X at pcut?
pts'discardpts'  //y- rewrite pts
?pts-dts'tpf'(   //   is now a closed GOP?
closed_gop'1') ) //   y- set closed GOP [note2]
))
put'target'end
```

[note1] Frame count (& sync) are retained.

[note2] MPEG's definition of the 'closed_gop' tag is vague regarding whether it is set in the preceding GOP: PTSes ..B B P B B, or the current GOP: PTSes I B B P...

See ***stride-match details***. *See* ***?-test details***.

The dts & pts & tpf properties are exact, but the dts & pts methods truncate when they write MPEG DTS & PTS tags.