



COLLABORA

Zero-Copy Pipelines in GStreamer

by Nicolas Dufresne

IRC: ndufresne

Email: nicolas.dufresne@collabora.com



Open First

Overview

- What does zero-copy mean ?
- Why do we copy ?
- What are the side effects ?



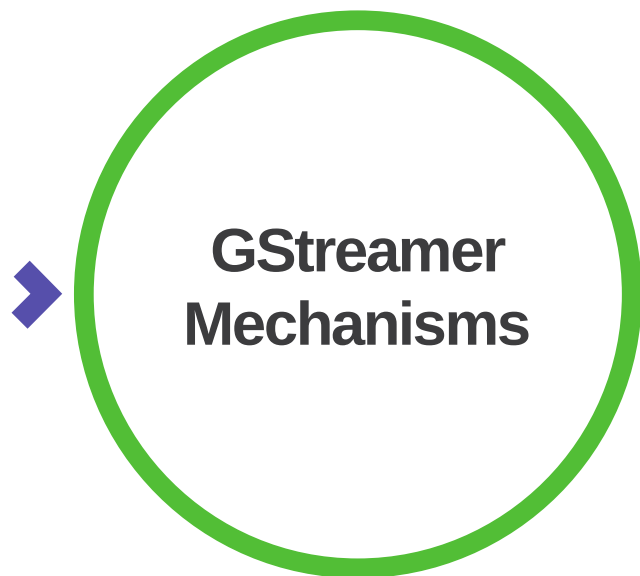
COLLABORA

What does zero-copy mean?



Definition

“Two Elements are doing zero-copy when the data produced by one Element is used by the other Element without requiring an idempotent transform.”



**BufferPools
and / or
Allocators**

**Memory
Logical
Grouping**

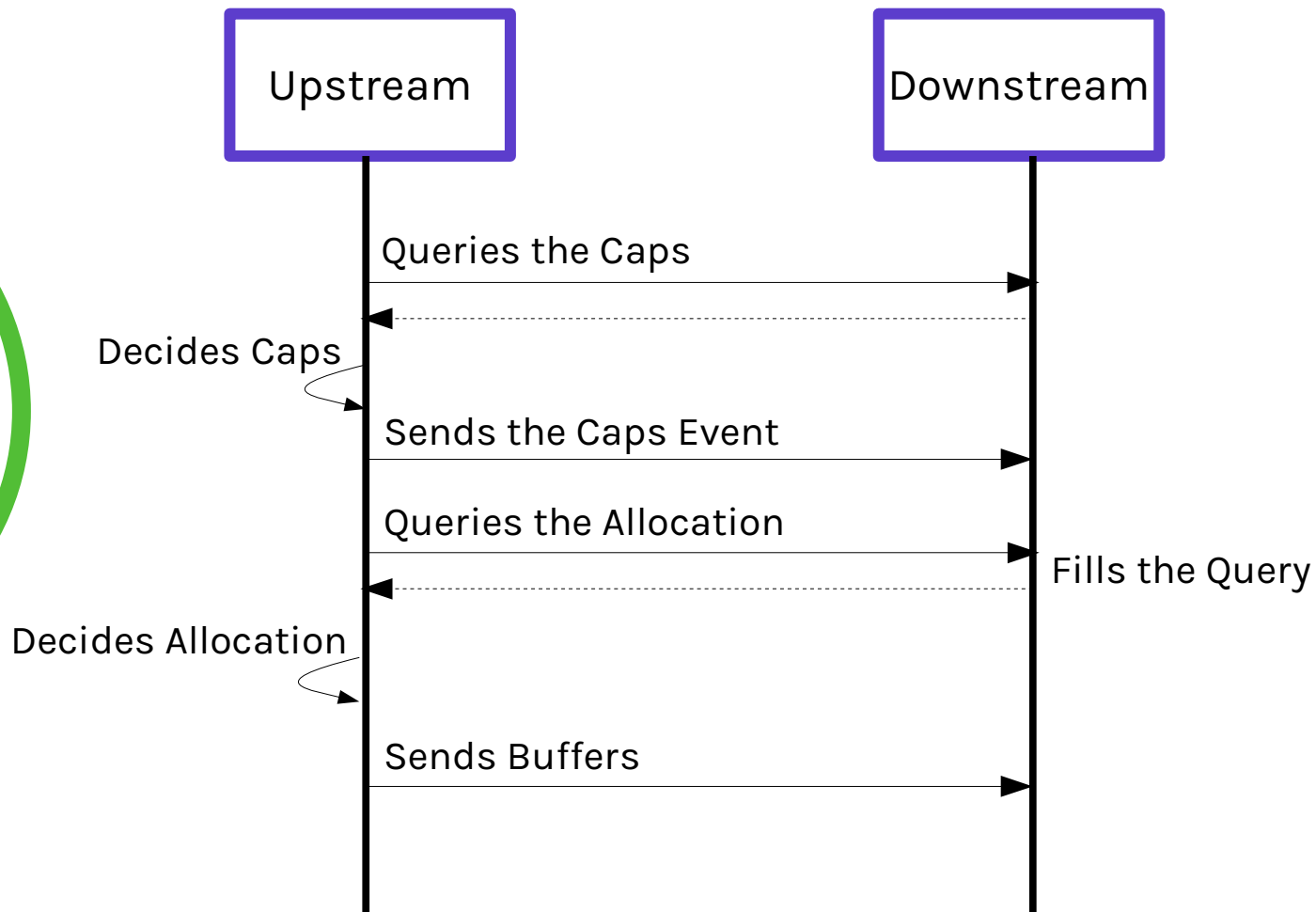
**Allocation
Query**

**Refcounted
Buffers and
Memories**

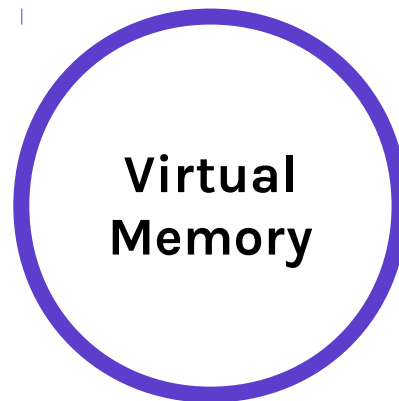
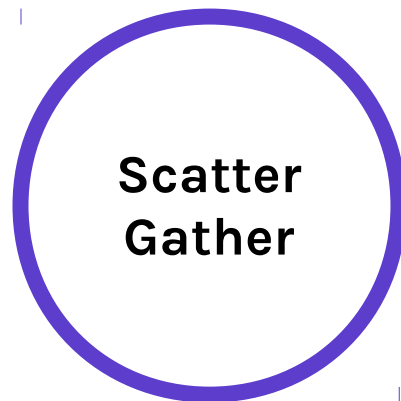
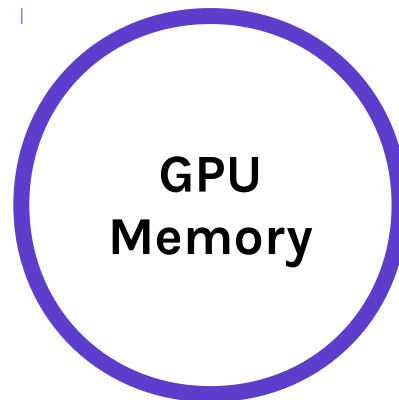
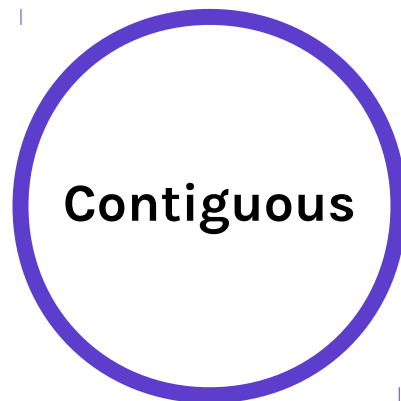
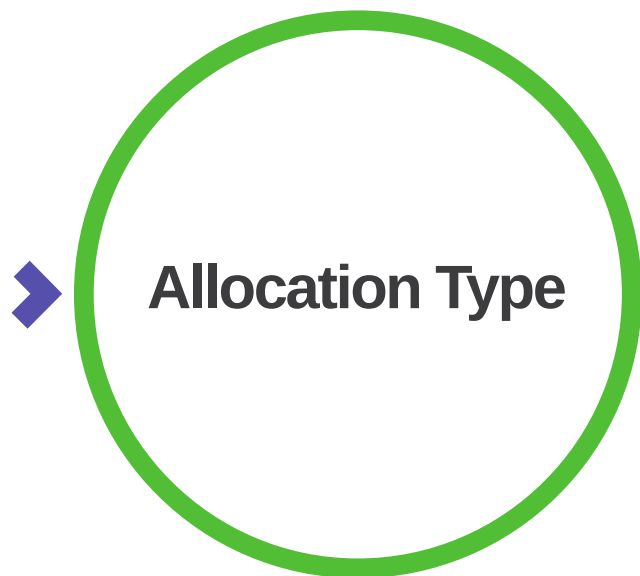
**Caps
Features
(Since 1.2)**

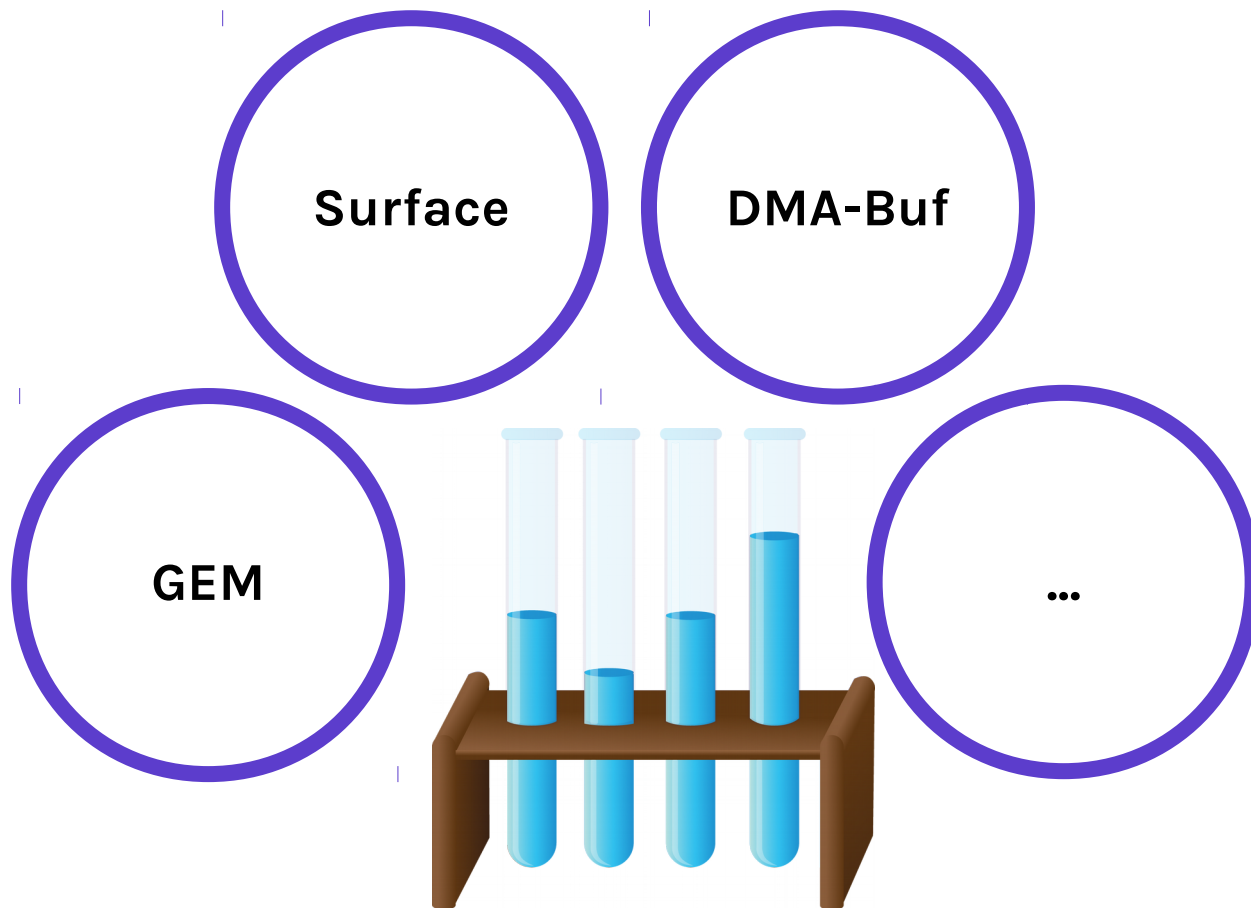


Upstream vs Downstream Allocation



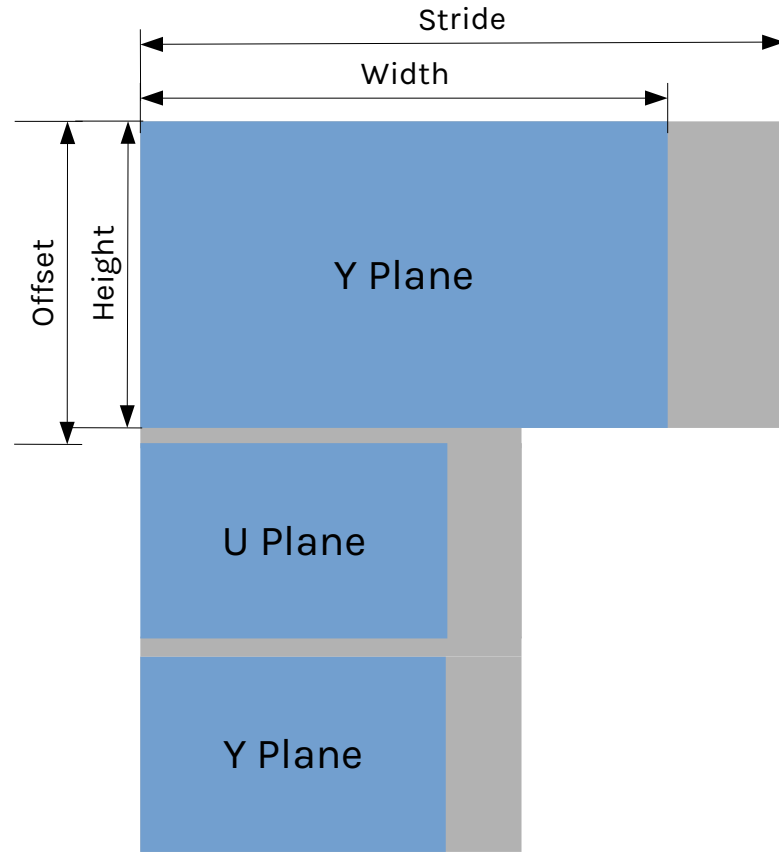
Why do we copy ?

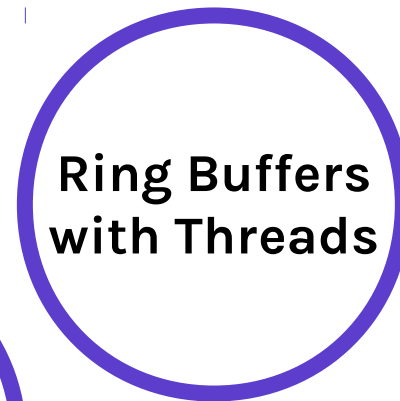
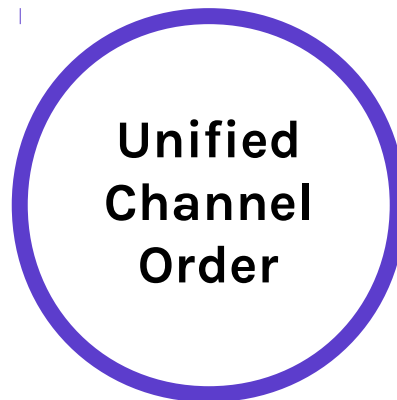
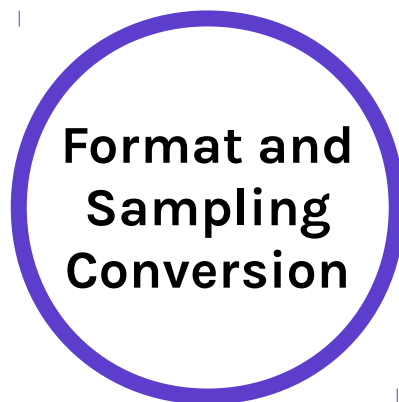


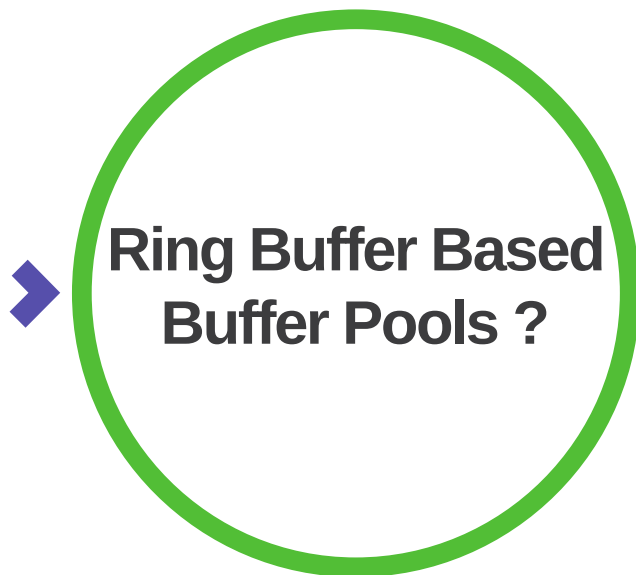




Video Alignment

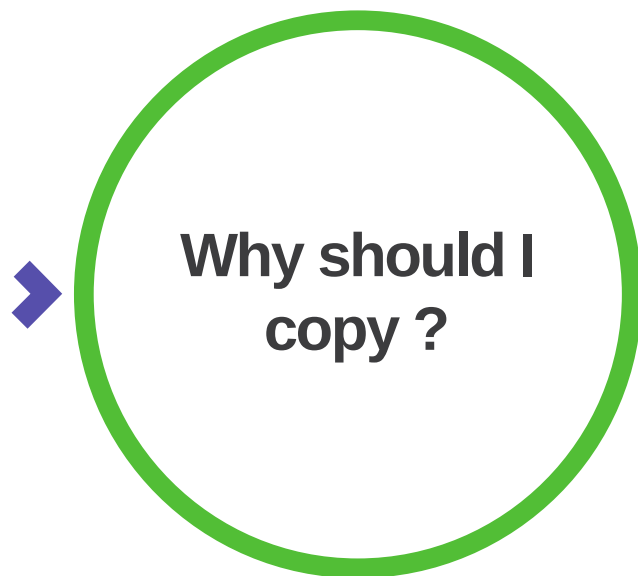




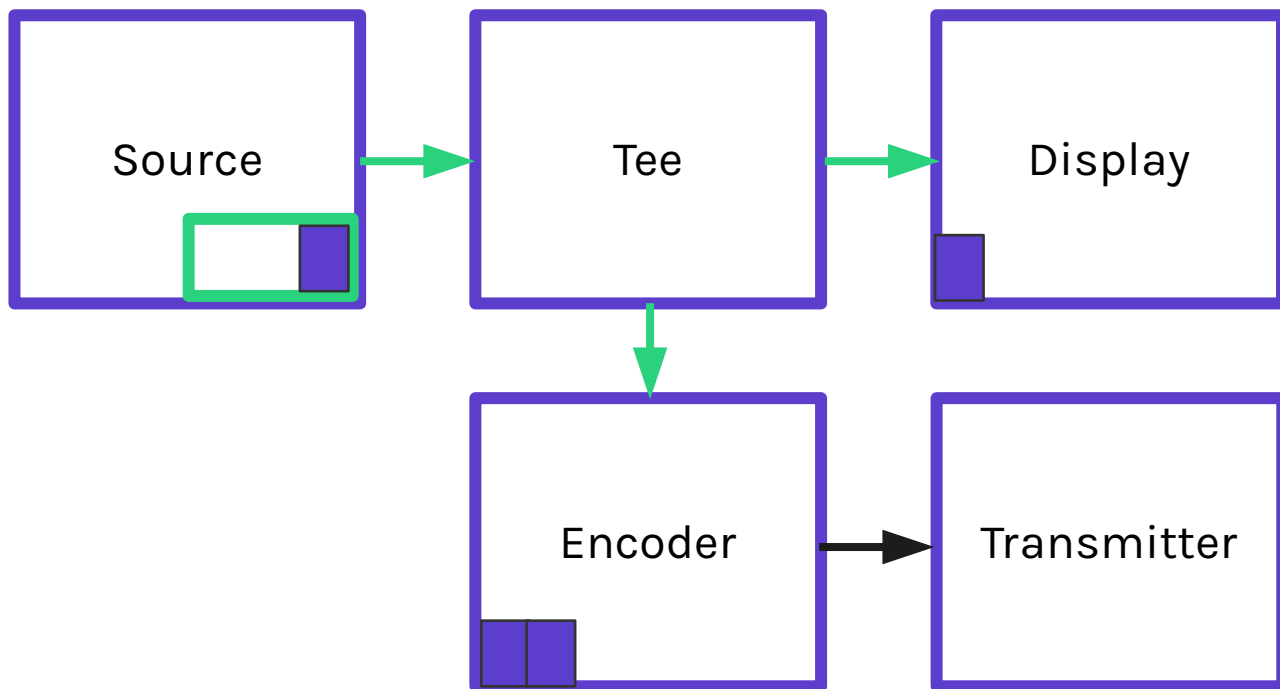
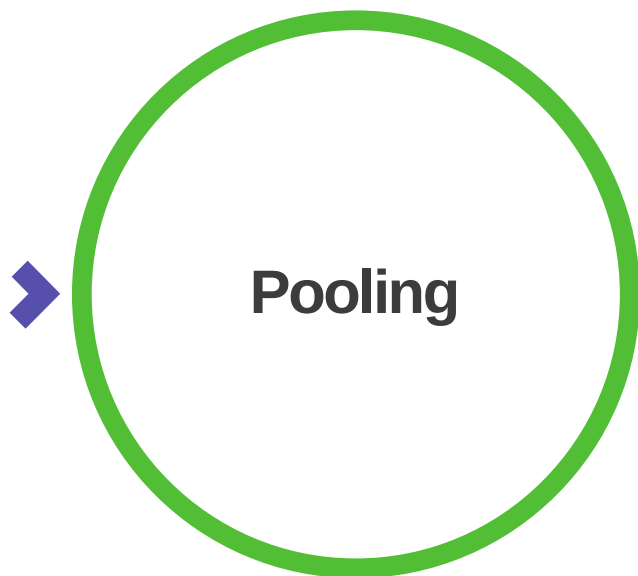


```
GstFlowReturn
gst_buffer_pool_acquire_buffer
    (GstBufferPool *pool,
     GstBuffer **buffer,
     GstBufferPoolAcquireParams *params);

struct GstBufferPoolAcquireParams {
    GstFormat                format;
    gint64                  start;
    gint64                  stop;
    GstBufferPoolAcquireFlags flags;
};
```



What are the side effects ?





Buffer Reclaiming

Upstream

Downstream

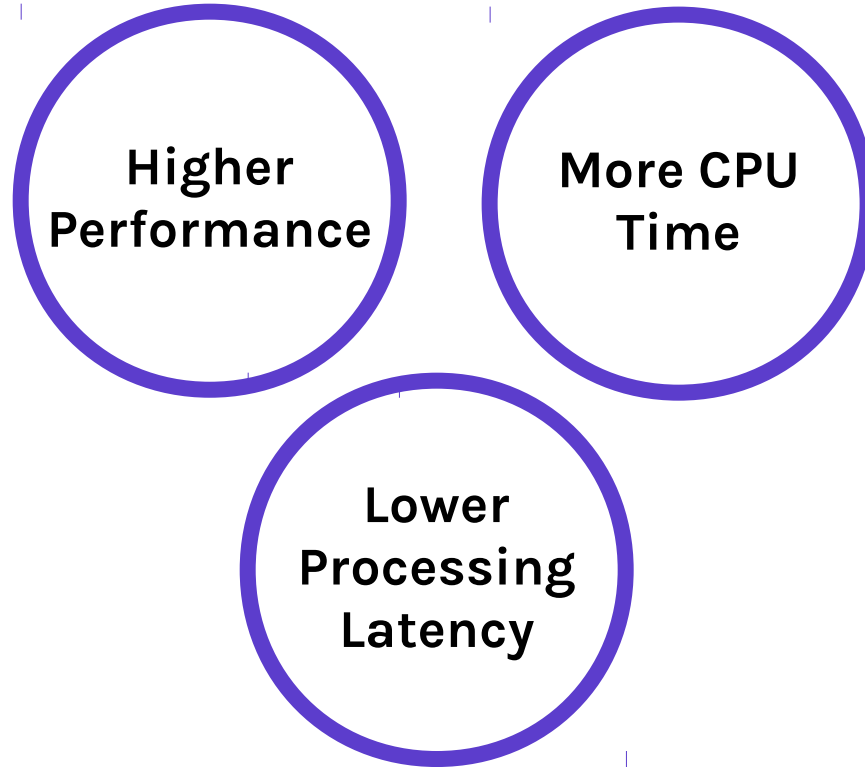
Allocation or Drain Query

Drains and returns ref frames
or
Copy and return ref frame

Mostly
Needed for
legacy APIs

Required
with V4L2
and OMX

Should not
be needed
for DMABuf
(it's a bug!)





Thank you!

