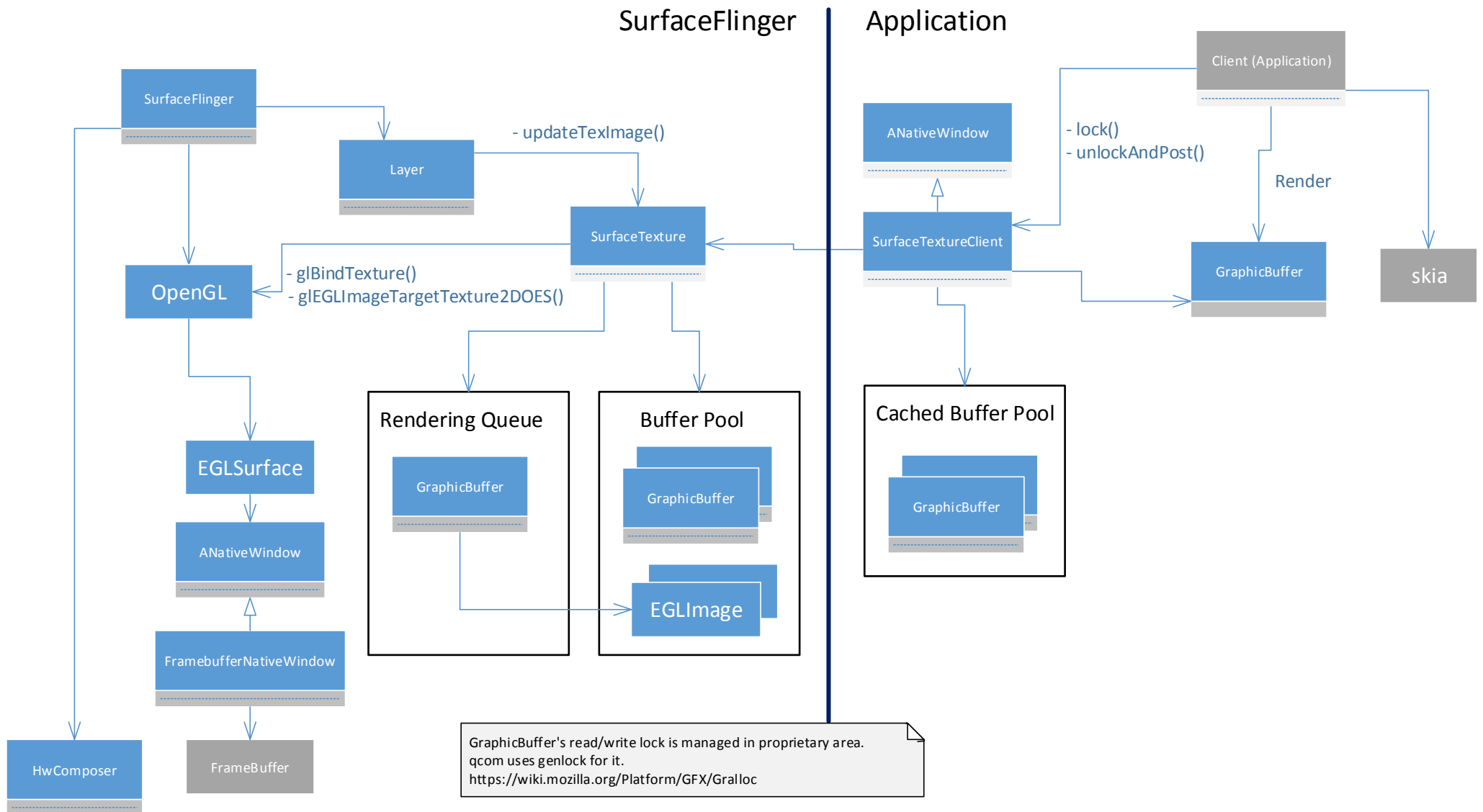
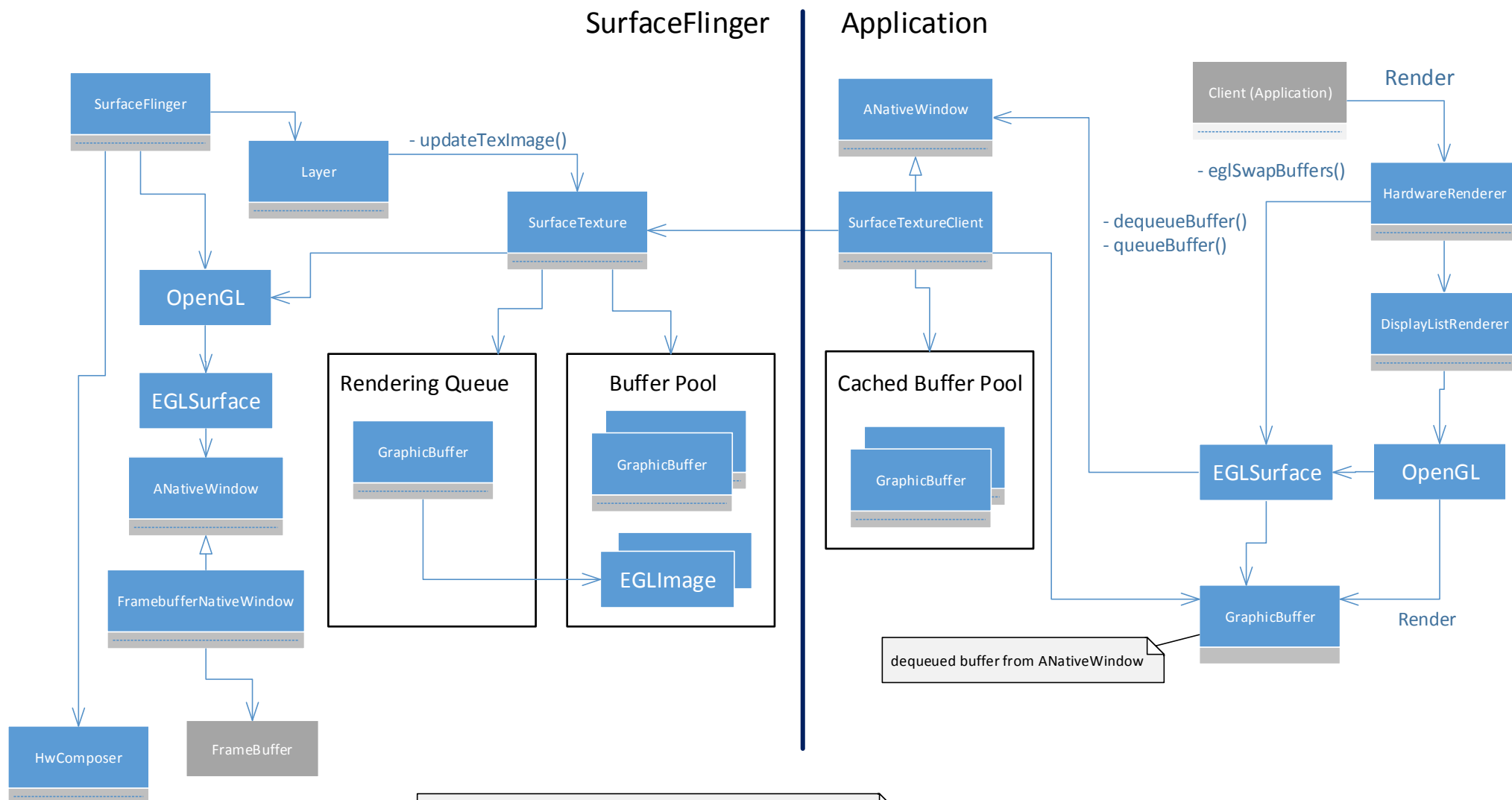


Android ICS: application rendering without OpenGL acceleration

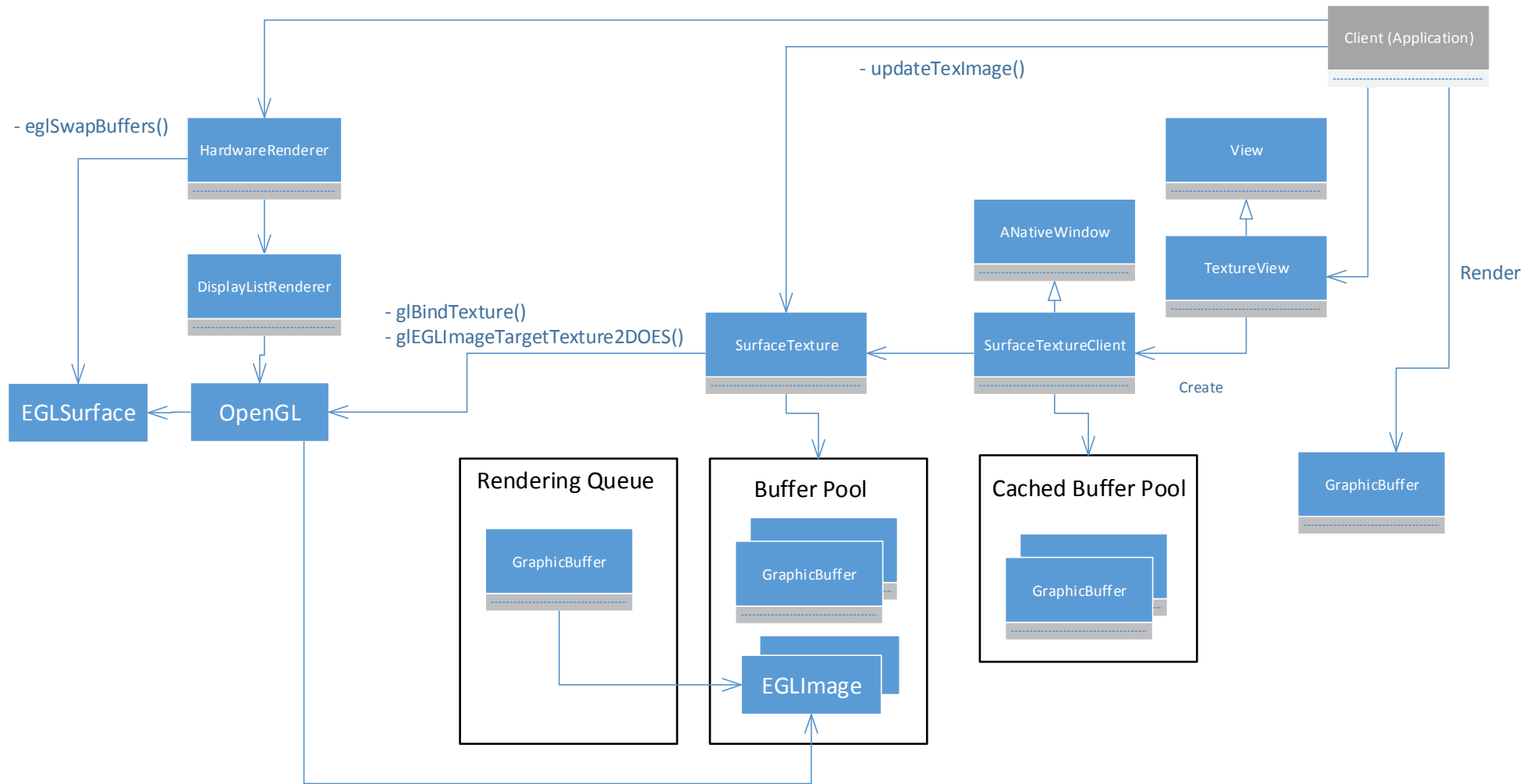


Android ICS: Application UI rendering with OpenGL acceleration



GraphicBuffer's read/write lock is managed in proprietary area.
qcom uses genlock for it.
<https://wiki.mozilla.org/Platform/GFX/Gralloc>

Android ICS: TextureView



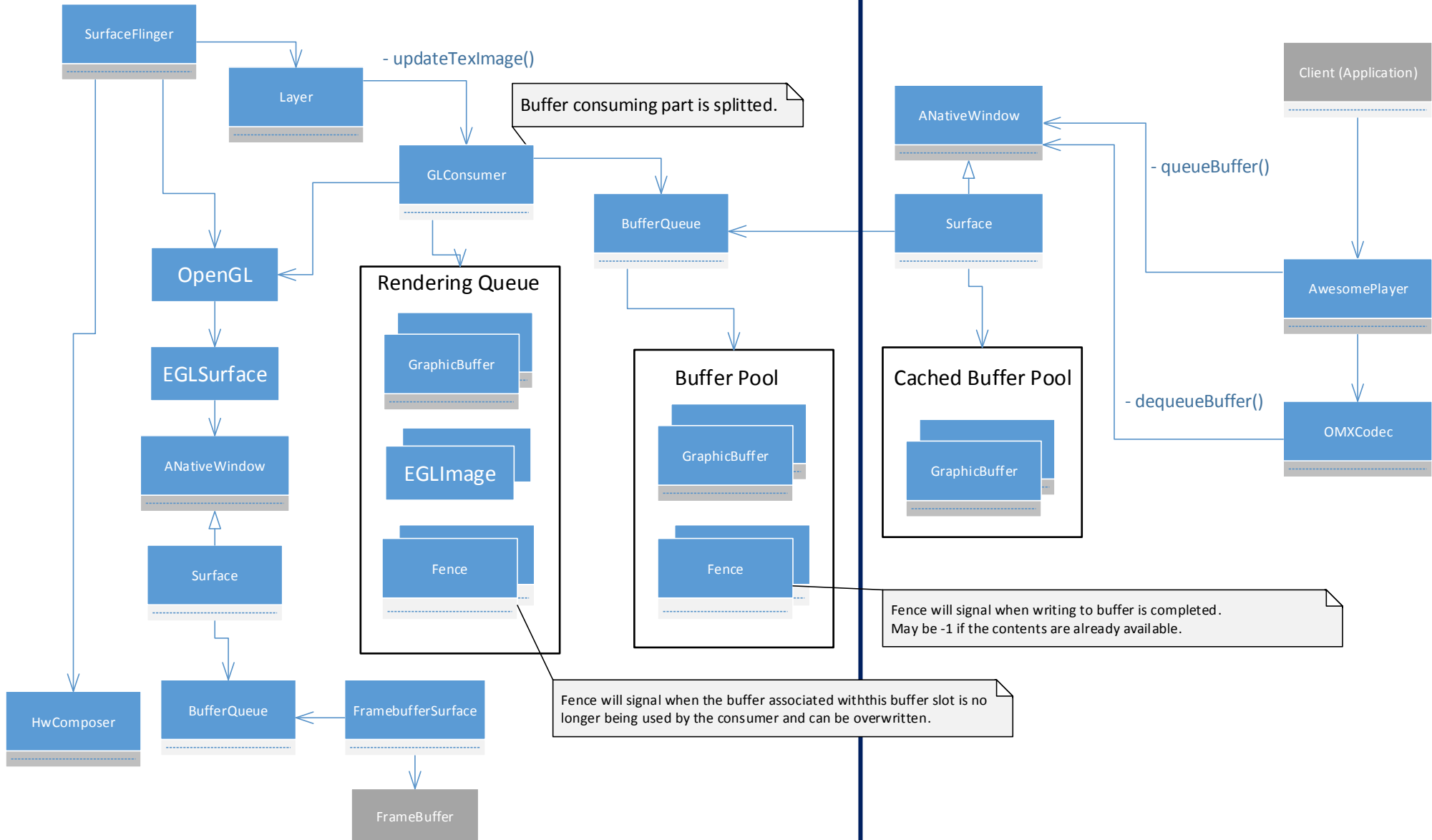
Android ICS: Video Playback



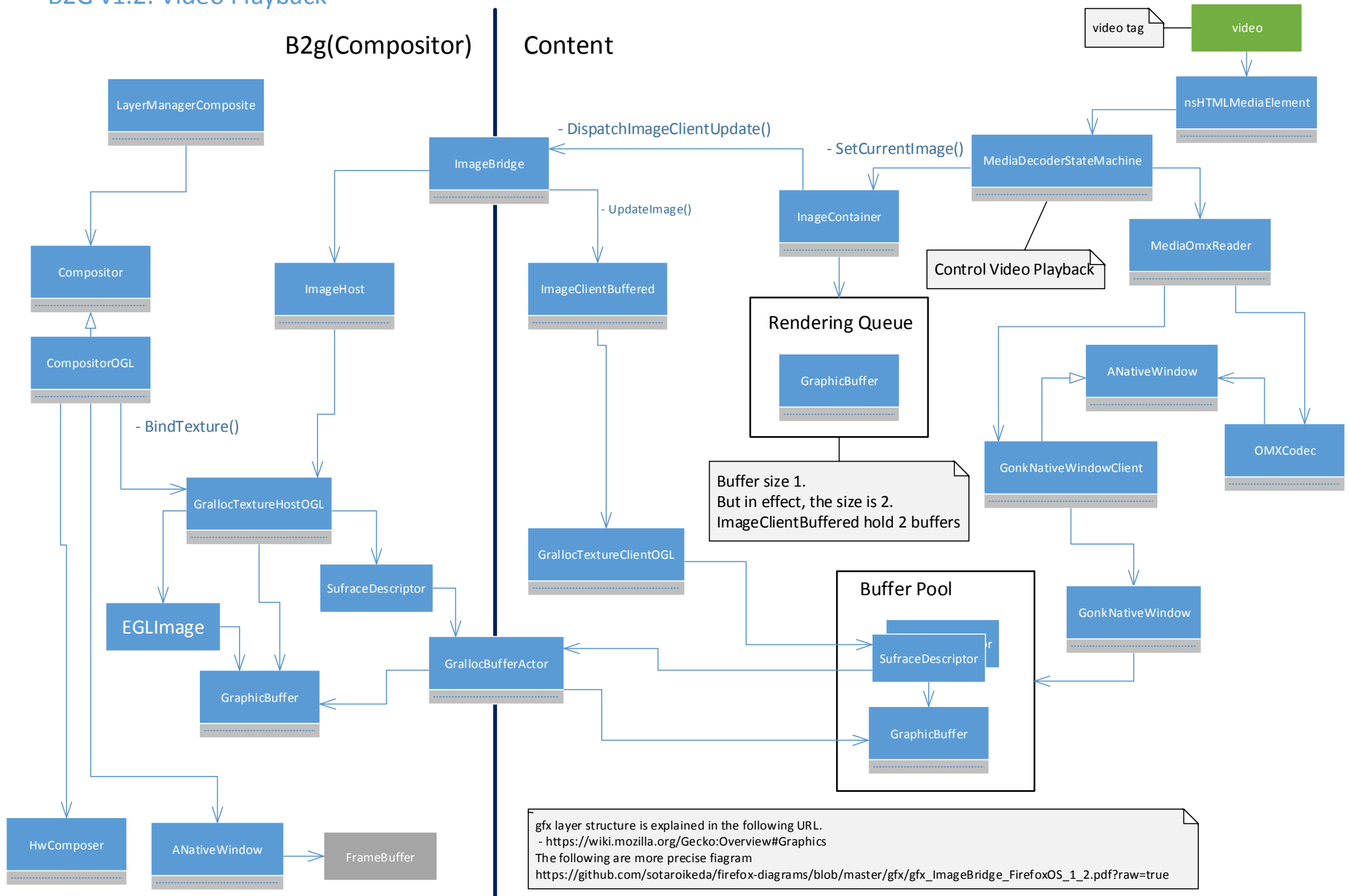
Android JB4.3: Video Playback

SurfaceFlinger

Application

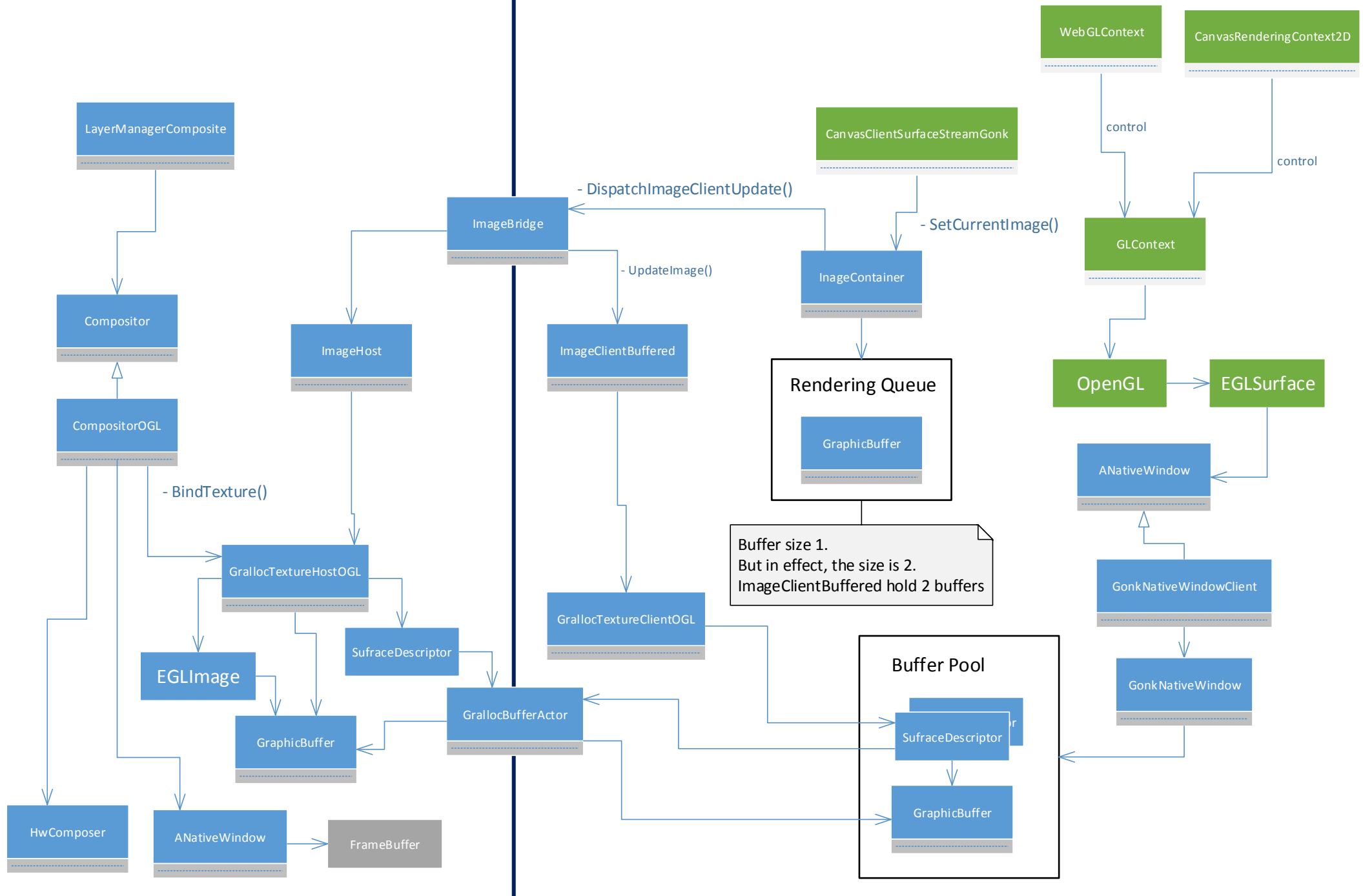


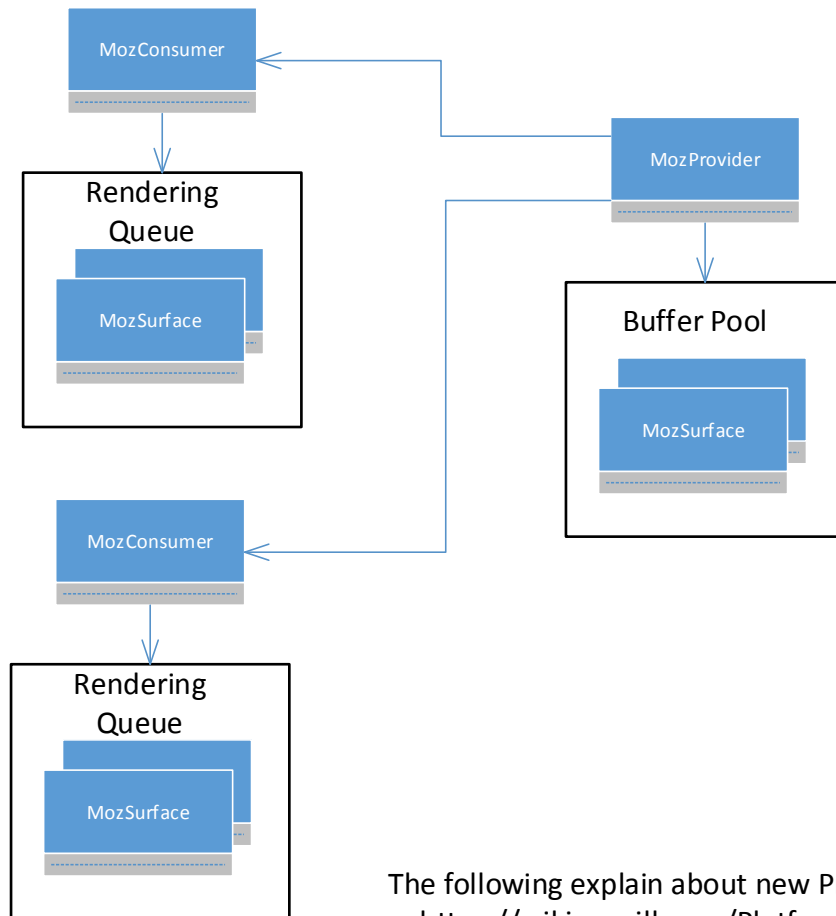
B2G v1.2: Video Playback



B2g(Compositor) Content

B2G v1.3: WebGL/Canvas(ICS gonk)





- [1] Simplify the model
 - + like EGLStream and SurfaceTexture
- [2] Use same code for in process and across process
 - + Unify “gfx layer” and “SurfaceStream”
- [3] Limit SurfaceDescriptor only around IPC code
- [4] Use ANativeWindow more if we could get more performance.
 - + WebGL/Canvas
- [5] Support multiple rendering target at one video stream
 - + current b2g does not support it.
 - + when a gralloc buffer is bounded to a texture, genlock write lock seems to be hold.
- [6] Support android::Fence
 - + It is present since android JB. It replaces genlock
 - + video rendering on b2g(JB gonk) have a problem
- [7] Remove retireFenceFd Fence wait
 - + Fence for Hw composer’s rendering complete
 - + android do not use it.
 - + nexus-4 wait here around 10ms
- [8] more robust to IPC failure/IPC shutdown

The following explain about new Provider/Consumer model
 - <https://wiki.mozilla.org/Platform/GFX/Surfaces>