

1 - DELIVERY MODEL

- SDR-only delivery (streaming, conferencing, web) *use Gamma (BT.1886)*
- On-demand premium HDR (Netflix-style VOD) *use PQ (SMPTE ST 2084)*
- Live HDR broadcast (sports, news, events) *use HLG (ARIB STD-B67)*
- Mixed audience: some HDR sets, some SDR sets *HLG handles both with one signal*

2 - CONTENT TYPE

- Cinematic content with precise artistic intent for brightness *PQ - absolute brightness control*
- Live broadcast with no pre-grading possible *HLG - relative brightness auto-scales*
- Talking heads, surveillance, screen-share *gamma is enough*
- Dolby Vision delivery target *PQ + DV dynamic metadata*

3 - TARGET DEVICES

- Target audience has HDR-capable TVs / phones (2020+) *PQ or HLG safe*
- Significant share of audience still on SDR-only devices *HLG keeps SDR fallback automatic*
- Tested tone-mapping on at least 3 HDR TV brands *PQ tone-mapping varies by player*
- Player negotiates HDR via EME / MediaCapabilities API *verify the tag reaches the decoder*

4 - METADATA TAGGING

- FFmpeg encode uses `-color_trc bt709 / smpte2084 / arib-std-b67` *match the chosen recipe*
- Verified with `ffprobe` that `color_transfer` is set after encode *trust nothing, verify*
- Tag survives packaging step (HLS / DASH / CMAF) *common drop point*
- CDN preserves metadata headers (`mdcv`, `clli` for HDR) *edge transforms can drop atoms*

5 - PIPELINE CONSISTENCY

- Bit depth matches the transfer function (10-bit min for PQ/HLG) *PQ at 8-bit produces severe banding*
- Colour primaries set to `bt2020` for HDR *primaries + transfer must agree*
- No transcode step strips the transfer-function tag *audit each stage with `ffprobe`*

MOST COMMON TRANSFER-FUNCTION BUGS IN PRODUCTION

- * HDR file tagged as SDR - player applies gamma to PQ data; oversaturated, neon greens, burning reds.
- * SDR file tagged as HDR - player applies PQ to gamma data; grey, dim, washed-out picture.
- * Tag dropped mid-pipeline - transcode strips metadata, next stage defaults to gamma, quality degrades silently.