HDR RENDERING ON NVIDIA GPUS

Thomas J. True, May 8, 2017

AGENDA

HDR Overview

Visible Color & Colorspaces

NVIDIA Display Pipeline

Tone Mapping

Programing for HDR

Best Practices

Conclusion

Q & A

HDR OVERVIEW

WHAT IS HIGH DYNAMIC RANGE?

HDR is considered a combination of:

- Bright display: 750 cm/m^2 minimum, 1000-10,000 cd/m^2 highlights
- Deep blacks: Contrast of 50k:1 or better
- 4K or higher resolution
- Wide color gamut

What's a nit?

A measure of light emitted per unit area.

1 nit (nt) = 1 candela / m^2

BENEFITS OF HDR

Tell a Better Story with Improved Visuals

Richer colors

Realistic highlights

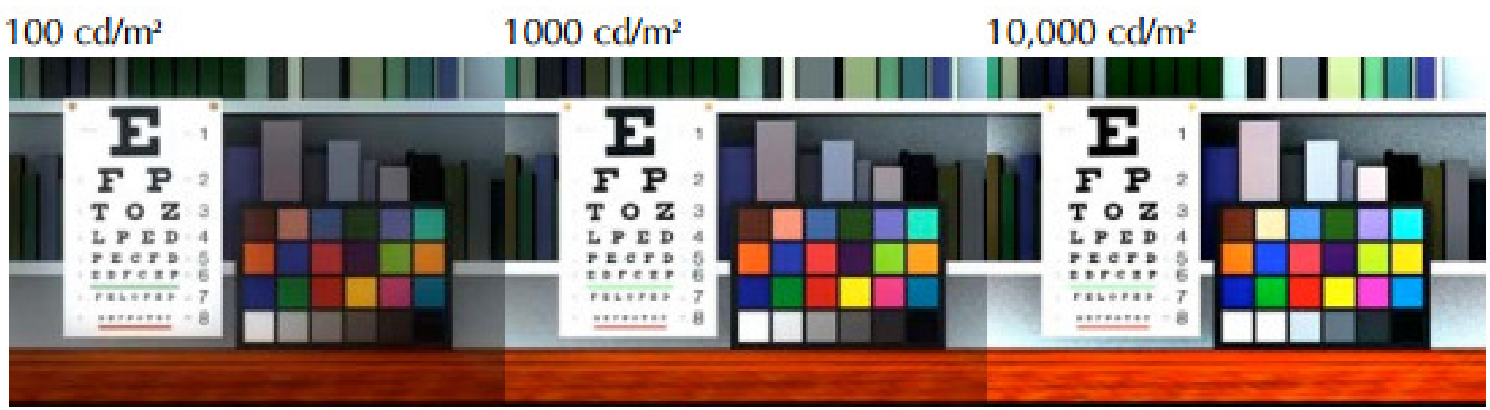
More contrast and detail in shadows

Reduces / Eliminates clipping and compression issues

HDR isn’t simply about making brighter images

HUNT EFFECT

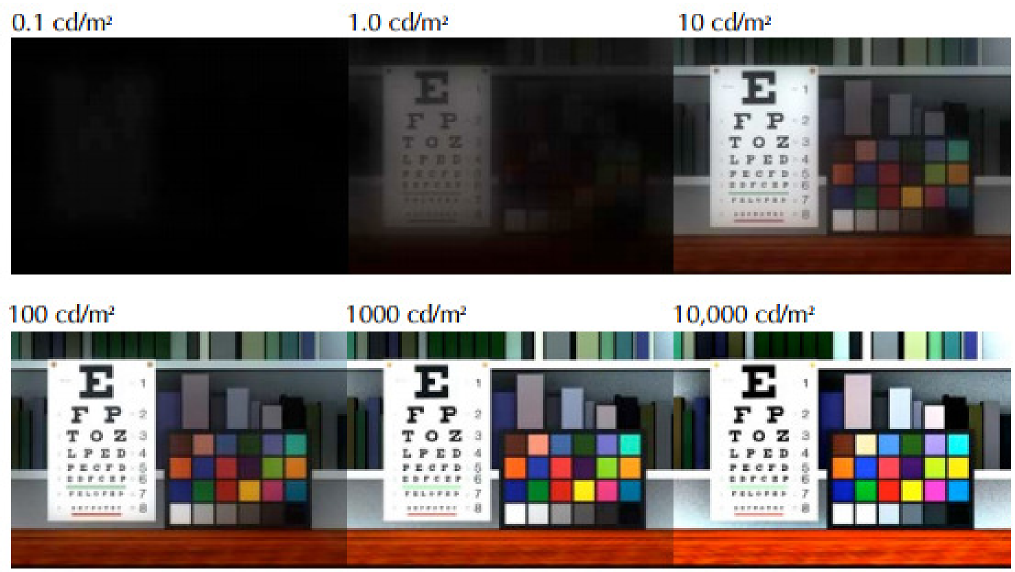
Increasing the Luminance Increases the Colorfulness



- By increasing luminance it is possible to show highly saturated colors without using highly saturated RGB color primaries
- Note: you can easily see the effect but CIE xy values stay the same

STEPHEN EFFECT

Increased Spatial Resolution



More visual acuity with increased luminance. Simple experiment – look at book page indoors and then walk with a book into sunlight

HOW HDR IS DELIVERED TODAY

Displays and Connections

High-End Professional Color Grading Displays – via SDI

- Dolby Pulsar (4000 nits)
- Dolby Maui
- SONY X300 (1000 nit OLED)
- Canon DP-V2420

UHD TVs – via HDMI 2.0a/b

- LG, SONY, Samsung… (1000 nits, high contrast, HDR10, Dolby Vision, etc)

Desktop Computer Displays – coming soon to a desktop / laptop near you

VISIBLE COLOR & COLORSPACES

REAL WORLD VISIBLE LUMINANCE RANGE

Range of 10^{17} Luminance Levels

$7.0 * 10^{10}$ cd/m^2	Lightning flash
$3.2 * 10^{9}$ cd/m^2	Sun (zenith)
$4.3 * 10^{5}$ cd/m^2	Sun (horizon)
$1.2 * 10^{5}$ cd/m^2	60W incandescent light bulb
$3.0 * 10^{4}$ cd/m^2	White paper in noon sunlight
$1.3 * 10^{4}$ cd/m^2	Clear sky (horizon)
$4.2 * 10^{3}$ cd/m^2	Full moon
$3.6 * 10^{3}$ cd/m^2	White paper in daylight shade
$1.3 * 10^{2}$ cd/m^2	White paper under office light
$1.0 * 10^{2}$ cd/m^2	White of computer monitor or TV
$1.0 * 10^{2}$ cd/m^2	Wax candle flame
$1.0 * 10^{2}$ cd/m^2	Clear sky, twilight
$2.4 * 10^{-1}$ cd/m^2	Brightest star (Sirius)
$1.3 * 10^{-3}$ cd/m^2	Absolute threshold (single flash)
$4.0 * 10^{-4}$ cd/m^2	Starless night sky
$7.5 * 10^{-7}$ cd/m^2	Absolute threshold (steady light)

REAL WORLD VISIBLE LUMINANCE RANGE

Human Visual Response

- Limited to 10^5 – 10^6 with a 95% contrast ratio of ~10000:1 (18 stops)
- Example: Full Moonlight – Can see details on the moon surface while simultaneously seeing details in the illuminated ground surface. (4200 cd/m2 to 0.012 cd/m2)
- Dark Adaptation
  - Slow, can take up to 30 minutes to see in the dark
- Light Adaptation
  - Fast, less than a second to a minute to adapt to bright light
- HDR displays should have a larger 10^7 dynamic range

BLACK

How Black Does HDR Need To Be?

Typical desktop LCD displays go down to about 0.1 – 0.3 cd/m^2

- In a dark room they clearly 'glow' when displaying black

Plasma displays have blacks of 0.002 cd/m^2

OLED displays have demonstrated blacks of 0.0001 cd/m^2

0.0001 cd/m^2 is *very dark*

- Requires 2 minutes of dark adaptation to see it

BRIGHTNESS

How Bright Does HDR Need To Be?

Typical desktop LCD displays are 100 – 350 cd/m^2

Digital cinema is 48 cd/m^2

User preference testing by Dolby has shown that 84% of viewers is satisfied with ‘highlights’ at 10,000 cd/m^2

10,000 cd/m^2 is the brightness you see when looking directly at a fluorescent tube – bright, but not painful

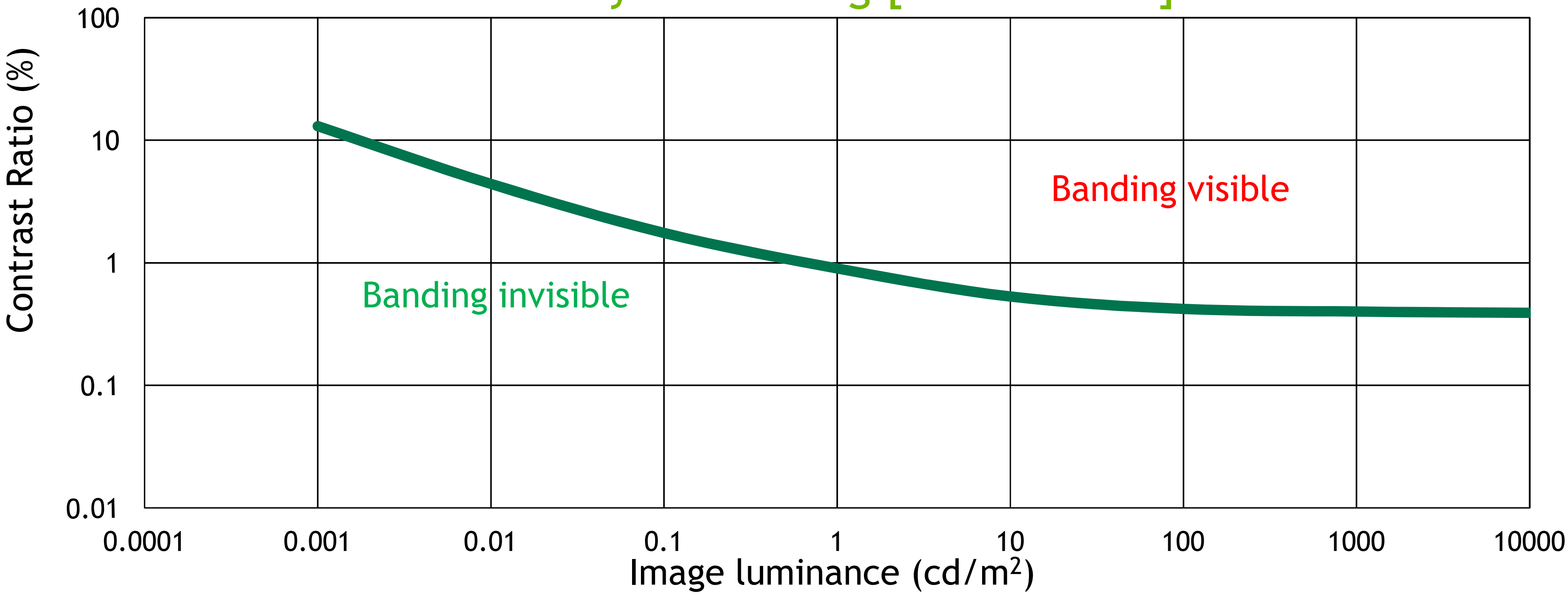
General consensus is that 750 cd/m^2 is a minimum for HDR, but that small areas of the screen going up to 1000 - 10,000 cd/m^2 is desirable

COLOR PRECISION

How do we avoid banding?

HUMAN PERCEPTION

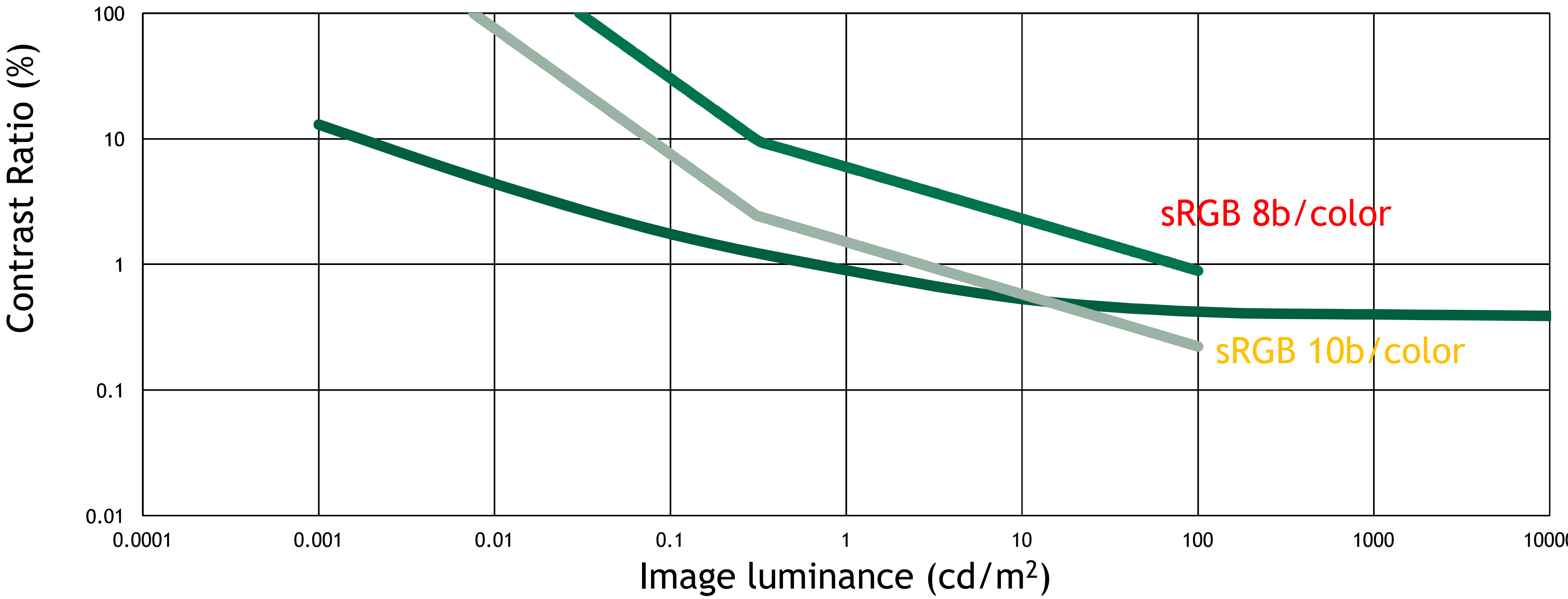
Visibility of banding [Barten 1999]



[Barten 1999] Peter G.J. Barten, ‘Contrast Sensitivity of the Human Eye and its Effects on Image Quality’

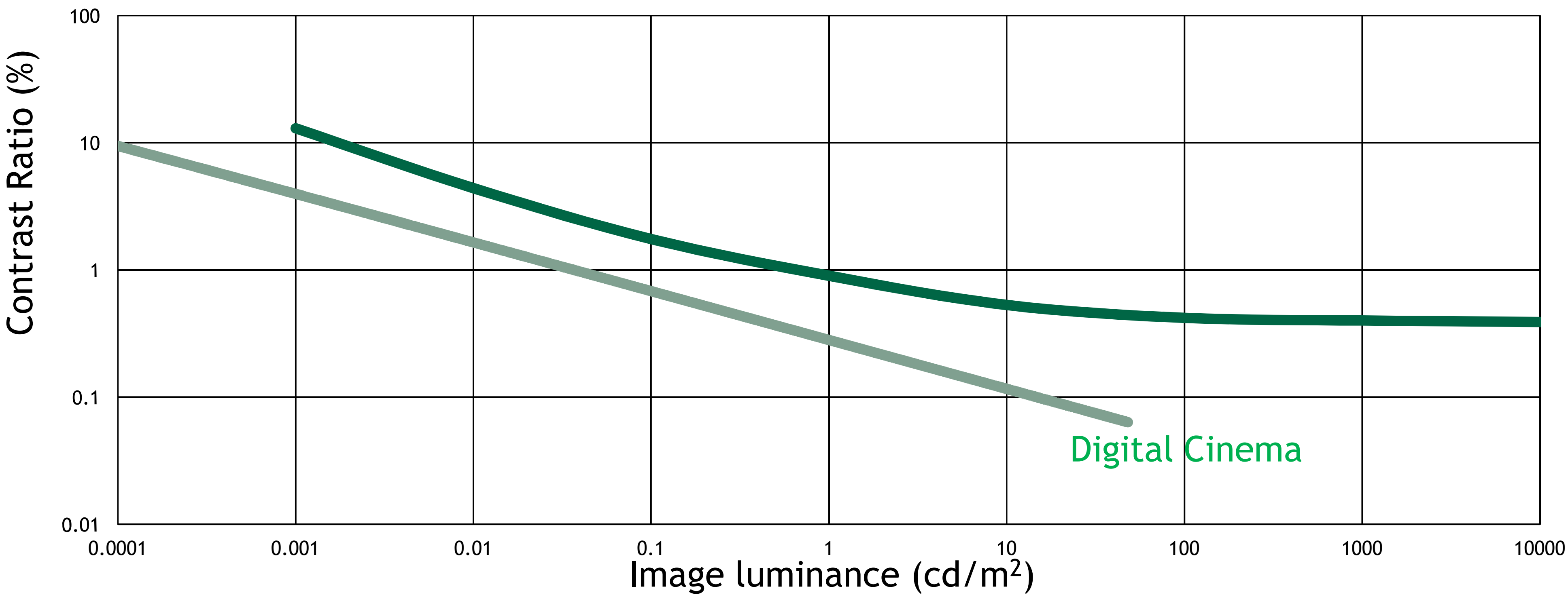
COLOR PRECISION

sRGB



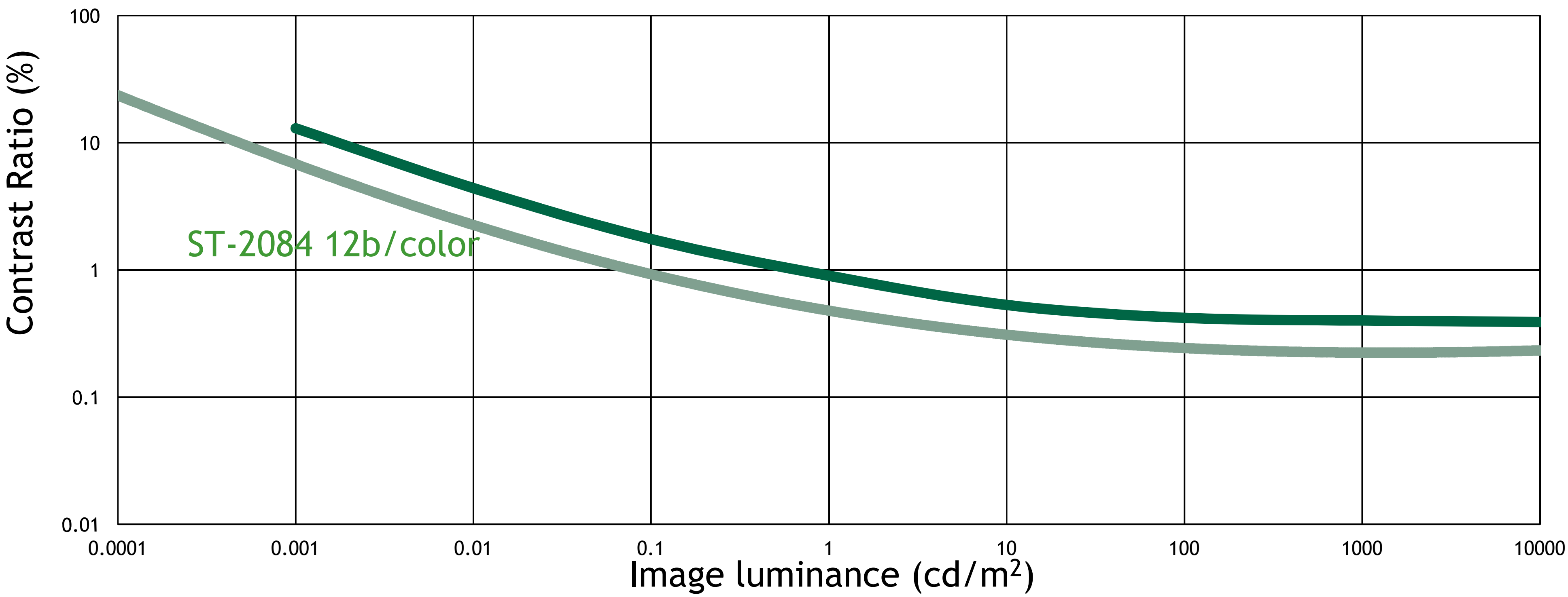
COLOR PRECISION

Digital Cinema - 12bit, gamma 2.6, full white = 48 cd/m^2



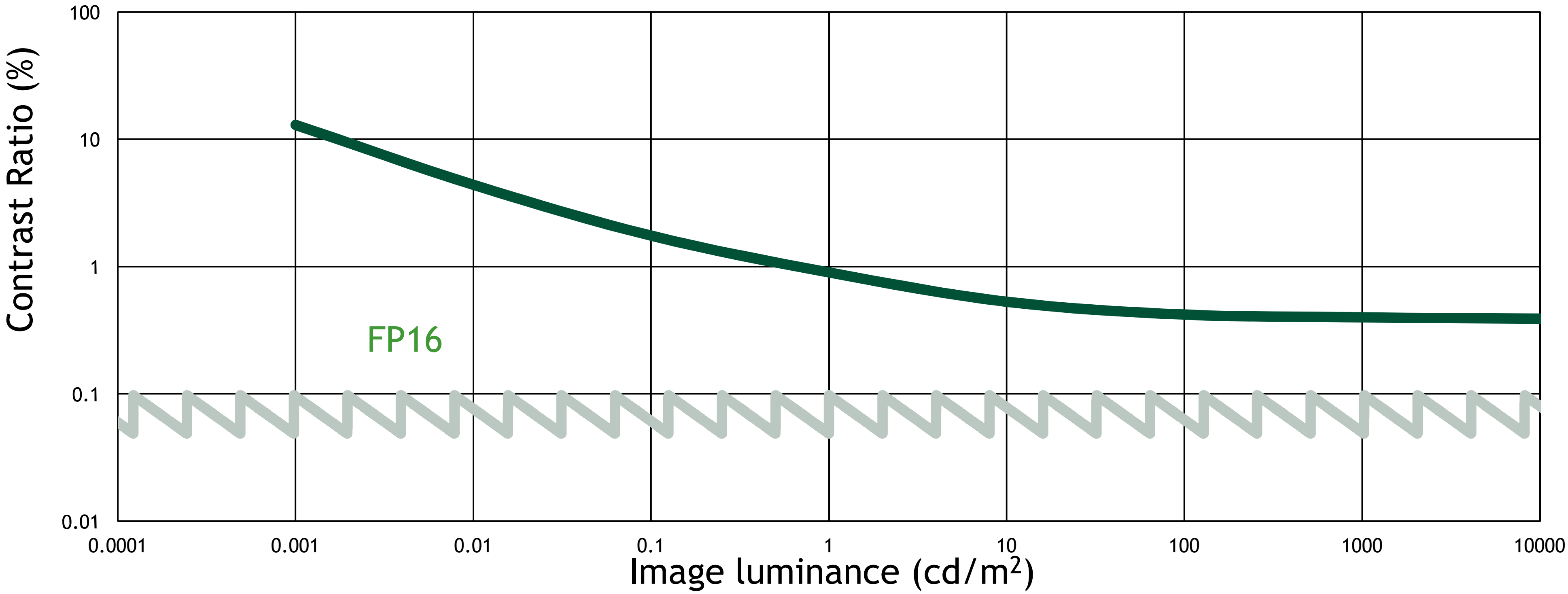
COLOR PRECISION

SMPTE ST-2084 - A new 12-bit HDR Transmission Standard



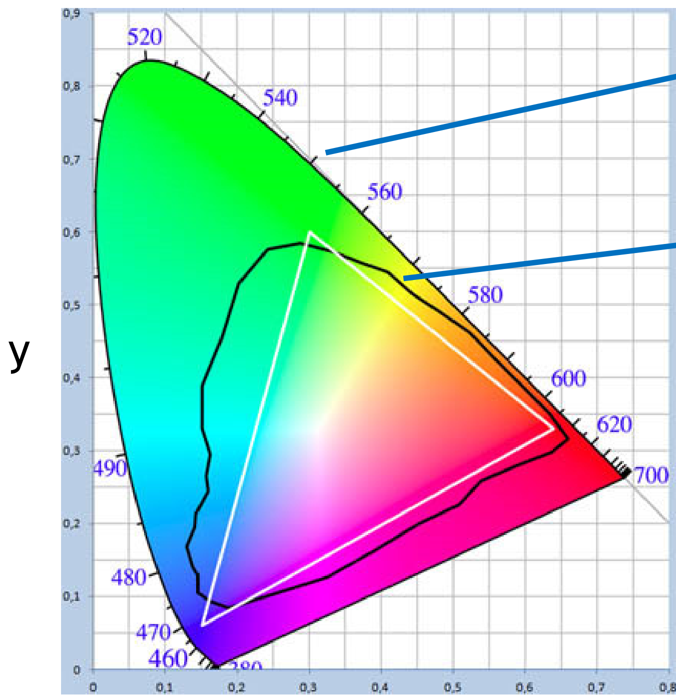
COLOR PRECISION

FP16 - For GPU Rendering



REAL WORLD VISIBLE COLORS

Pointer’s Gamut of Naturally Occurring Colors



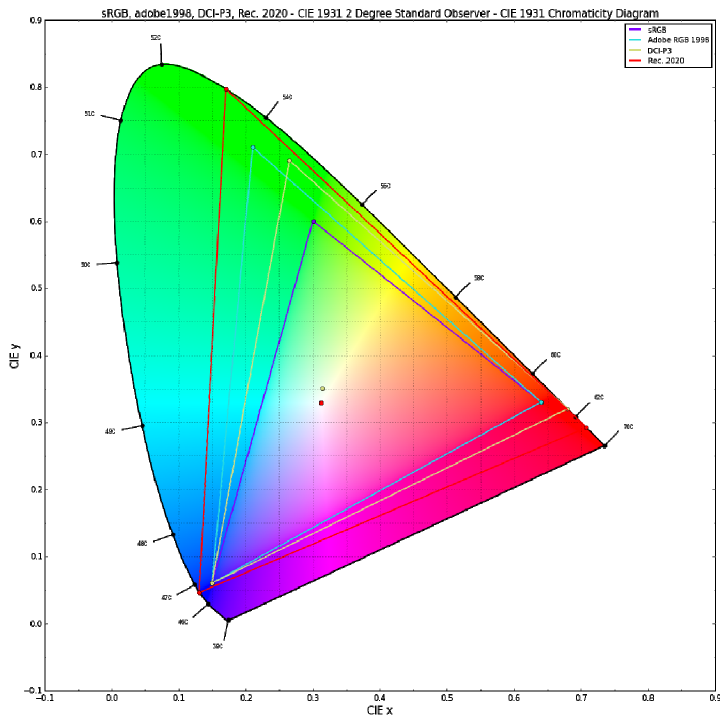
The CIE 1931 chart defines a coordinate system for all possible colors that the human eye can see

[Pointer 1980] the colors of ‘real world’ objects

[Pointer 1980] M.R. Pointer, ‘the Gamut of Real Surface Colors’

COLORSPACES

Comparison of Common Colorspaces



Rec 2020

- UHDTV Standard
- 60% of visible colors
- 99% of Pointer’s Gamut

DCI-P3

- Digital cinema projectors

AdobeRGB (1998)

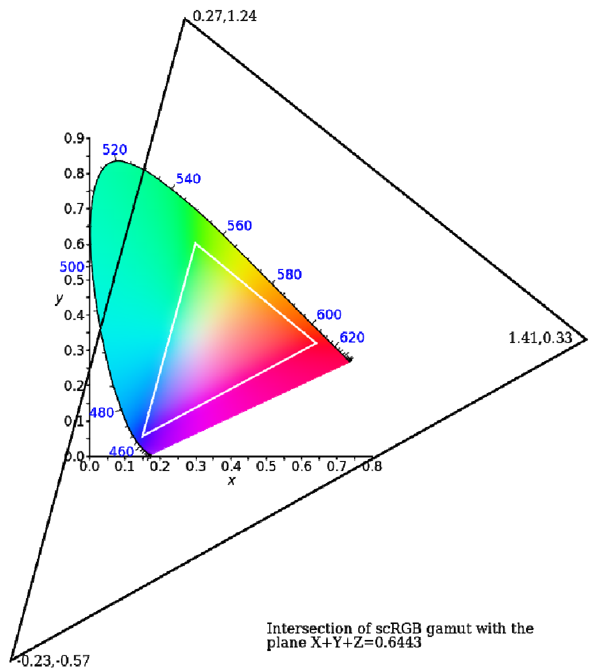
- Includes printable colors
- Same red and blue
- Purer green

sRGB (1996)

- Designed around CRT
- Same primaries as Rec. 709
- 33% of visible colors
- 70% of Pointer’s Gamut

COLORSPACES

Comparison of Common Colorspaces



sRGB (1996)

- Designed around CRT
- Same primaries as Rec. 709
- 33% of visible colors
- 70% of Pointer's Gamut

scRGB (Vista)

- [-0.5, 7.5]
- (1,1,1) matches sRGB white

https://en.wikipedia.org/w/index.php?curid=27214602

TONE MAPPING

COLORS

Scene Referred vs Output Referred

Scene Referred (Scene Linear)

- Linear colors as they represent light in the scene
- Photons striking the virtual film

Output Referred (Output Linear or Display Referred)

- Linear colors as they are represented by the display
- Photos emitted by the display
- May have an EOTF applied.

TONE MAPPING

Conversion from Scene Referred to Output Referred

Compresses or clips the color data into the output range

Compresses shadows and highlights

Enhances mid-tone contrast

Irreversible, data is lost

TONE MAPPING

Why Tone Map for HDR?

HDR displays still limited (1000 nit max)

Real world luminance is much higher

- Sun over 1000x more luminous
- 100w bulb over 10x more luminous

Permits differentiation of luminance levels

No one true tone mapper, choice depends on the desired aesthetics

For best results in HDR the tone mapper must understand the output luminance range of the display.

TONE MAPPING

Linear

Scale and clip to [0,1]

Same general problems as in LDR

- Hard clip at the limit of the capabilities of the display
- Sun and light bulb likely to have same luminance on screen

Image will be over compressed and look dull

Needs to account for the larger luminance range when scaling / clipping

- Otherwise, scene will just get brighter

TONE MAPPING

Reinhard

Classic x/(x+1)

No concept of output brightness

- In HDR, images just get a lot brighter

Example: 0.18 will change from 12-45 nits to 150+ nits

- 0.18 is often considered the color of asphalt after exposure
- Result is a bright road

Limited control

E. Reinhard, M. Start, P. Shirley and J. Ferwerda,
"Photographic Tone Reproduction for Digital Images"
https://www.cs.utah.edu/~reinhard/cdrom/tonemap.pdf

TONE MAPPING

Drago

Algorithmic operator(similar to Reinhard)

Compressed range using an adaptive log scale

Provides argument for display output luminance

Better adapts to display brightness

[Drago 2003] F. Drago, K. Myszkowski, T. Annen and N. Chiba, “Adaptive Logarithmic Mapping for Displaying Hight Contrast Scenes”, Eurographics 2003.
http://resources.mpi-inf.mpg.de/tmo/logmap/logmap.pdf

TONE MAPPING

Filmic

S-curve in logarithmic space

Enhances mid-tones

Compresses shadows and highlight

Approximates the behavior of traditional film

ACES

Academy Color Encoding System

Standard for digital post-production

Driven by the Academy of Motion Pictures

Provides framework for end-to-end processing and preservation of data

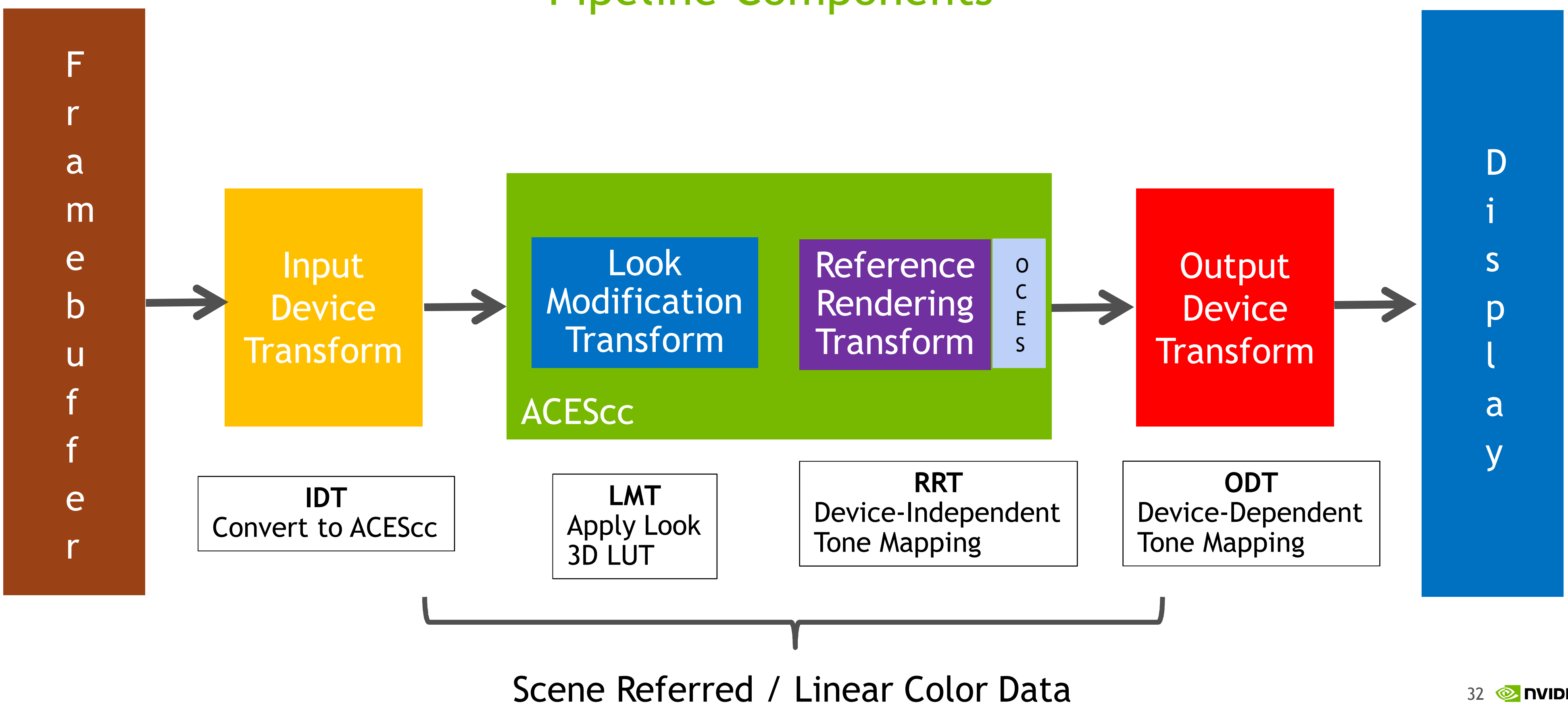
Defines reference transforms as part of the framework

Tone mapping for different classes of displays

https://github.com/ampas/aces-dev/releases/tag/v1.0.3

ACES

Pipeline Components



ACES

Tone Mapper

Tone mapper is a filmic sigmoid-style-curve

Defined by segmented quadratic spline in reference implementation

- Two splines joined at middle gray

Operates per-channel in a wide color space

- Results in natural desaturation at the shoulder

Input middle gray is set at 0.18

ACES

Parameterized ACES

Parameterized ODT developed by NVIDIA

Allows adaptation of the reference transforms to a wider set of uses

- Alter output middle gray level
- Alter input and output range of tone mapper
- Saturation adjustment
- Contrast adjustment

GAMUT MAPPING

Mapping of Unrepresentable Colors to Representable Colors

Stretching or compressing one color space to fit within another.

Remapping of the chromaticity values

Many different methods to remap the color space (clip, soft clip, scale, etc)

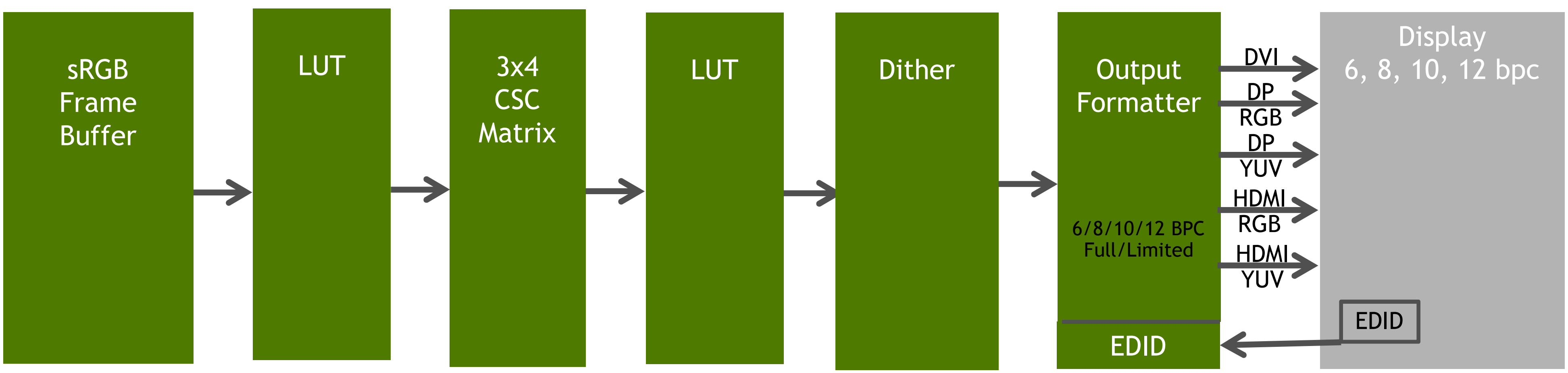
All methods have non-trivial caveats (hue shifts, memory colors, etc)

NVIDIA DISPLAY PIPELINE

TT1

TRADITIONAL DISPLAY PIPELINE

Function Block Diagram



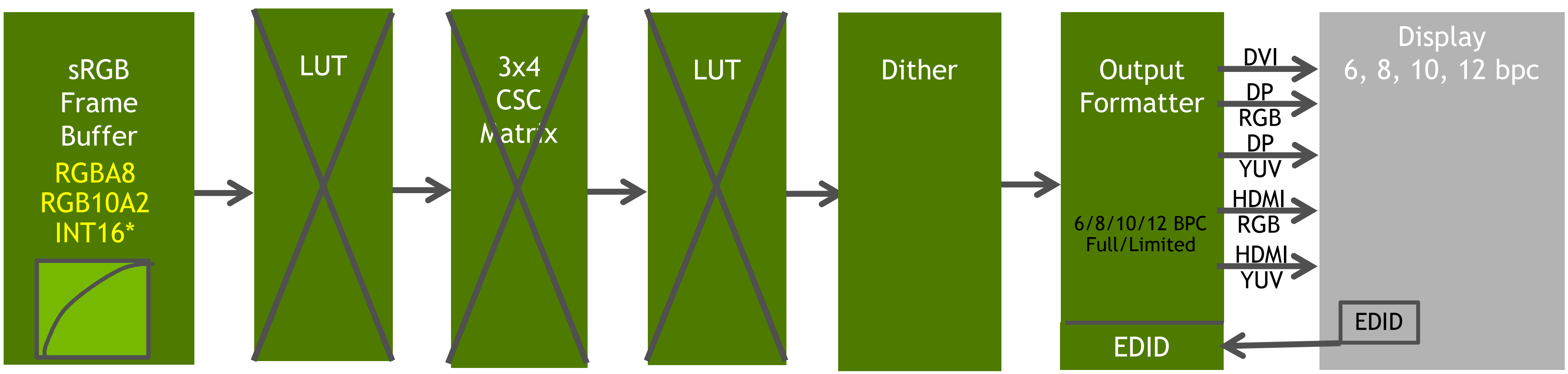
Slide 37

TT1 Where does RGB to YUV conversion happen for YUV displays -- DIsplay Engine?
Thomas True, 4/14/2017

TT1

TRADITIONAL DISPLAY PIPELINE

Case 1: Pass Through



sRGB(128,128,128) → (128,128,128)

Linear sRGB(128,128,128)

Adobe RGB(128,128,128)

sRGB(2048,2048,2048) → (2048,2048,2048)

*OpenGL + Pascal GPU Only

Slide 38

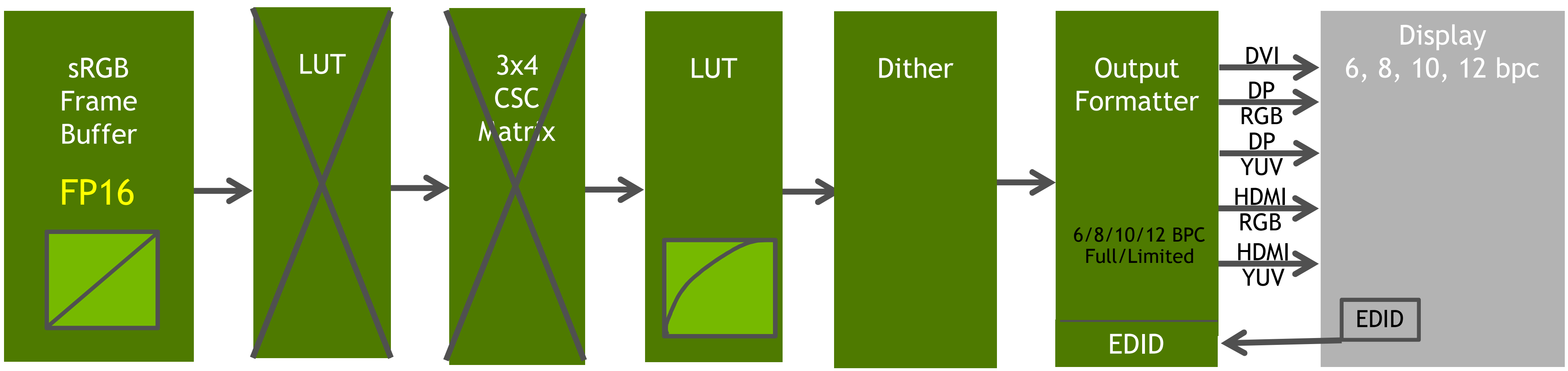
TT1 Where does RGB to YUV conversion happen for YUV displays -- DIsplay Engine?

Thomas True, 4/14/2017

TT1

TRADITIONAL DISPLAY PIPELINE

Case 2: Gamma Correction



Linear sRGB(0.5,0.5,0.5) → (228,228,228)

Slide 39

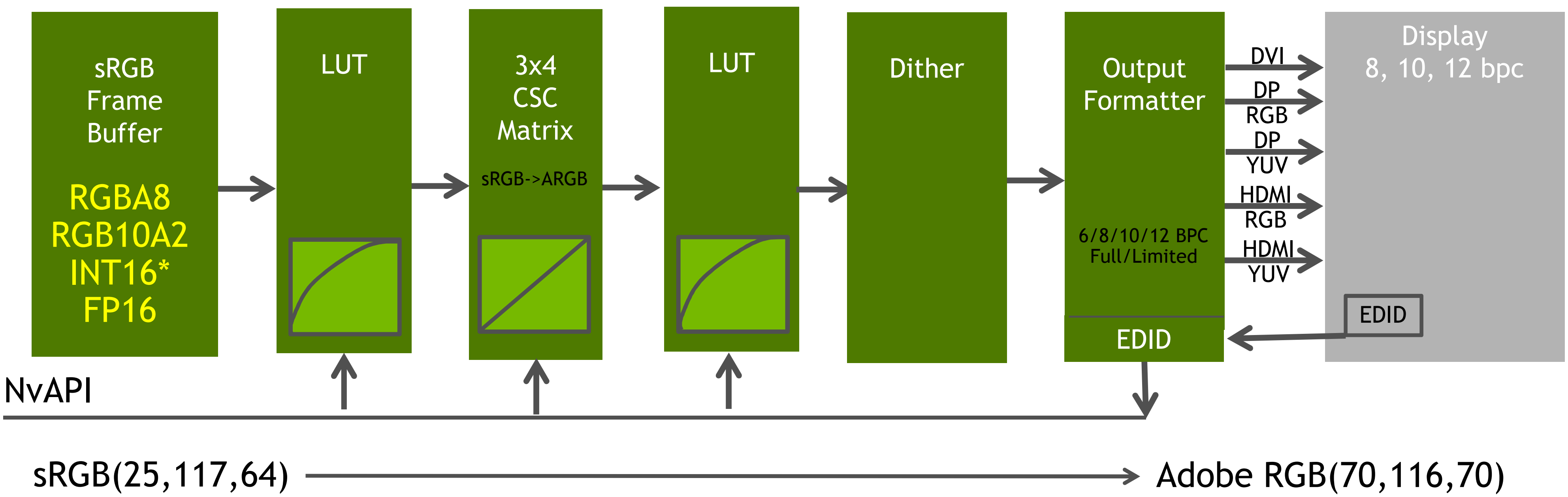
TT1 Where does RGB to YUV conversion happen for YUV displays -- DIsplay Engine?

Thomas True, 4/14/2017

TT1

TRADITIONAL DISPLAY PIPELINE

Case 3: Custom CSC



*OpenGL + Pascal GPU Only

Slide 40

TT1 Where does RGB to YUV conversion happen for YUV displays -- DIsplay Engine?

Thomas True, 4/14/2017

NVIDIA CSC SDK

Efficient RGB Color Conversions in the GPU Display Pipeline

- De-gamma and re-gamma 10-bit HW LUTs
- 4x3 Color Conversion Matrix
- Direct access to monitor colorimetry parameters from the EDID for direct construction of the CSC matrix required for display color correction.
- Build-in conversion matrices for standard RGB color spaces (sRGB, Adobe RGB, Apple RGB, NTSC, HDTV-Rec709, PAL, DCI-P3, Rec. 2020 + Rec. 709 EOTF, Rec. 2020 + SMPTE ST2084 PQ HDR EOTF)
- Controlled via NvAPI (NDA version)
- Fermi and later GPUs (Quadro and GeForce)

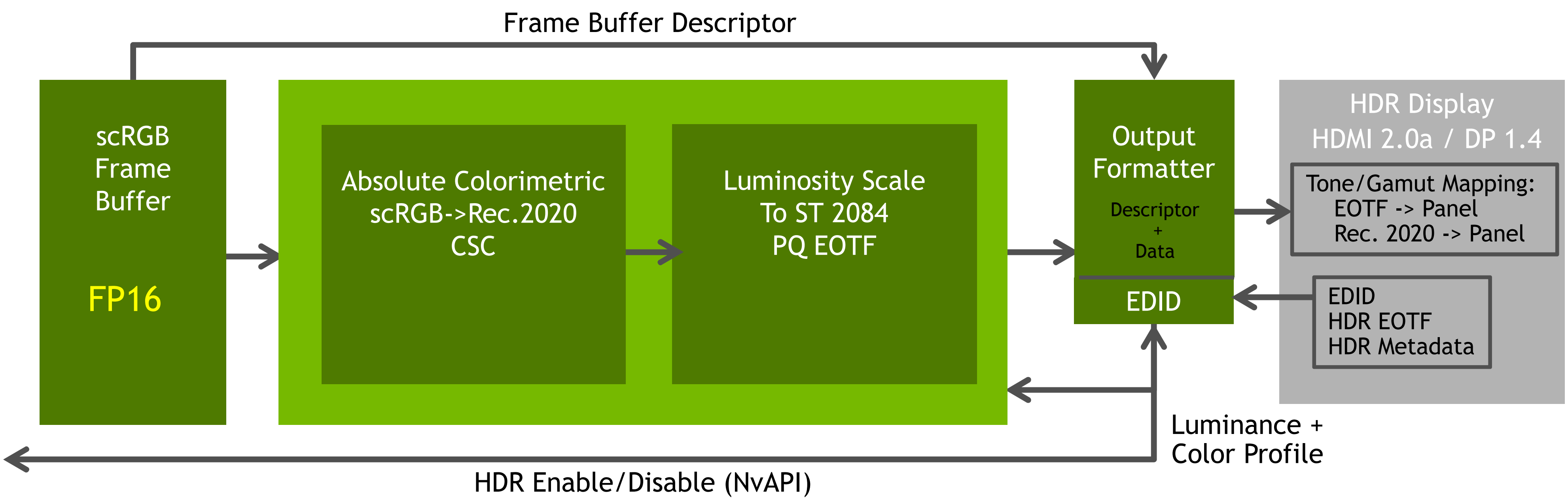
NVIDIA CSC SDK

Limitations

- Only available on Windows via NDA version of NVAPI
- Designed primarily for sRGB monitor color correction scenarios where source color space sRGB. In case source color space is not an sRGB, for example Adobe RGB, the default monitor color profile must be updated in the Windows color management settings to communicate the CSC configuration to other windows managed applications.
- Re-gamma LUT can easily be overridden by Win32 and DirectX, applications should read out and restore (push and pop) to ensure that the expected LUT is correct.
- Supports only matrix-based color conversions. This maybe insufficient for color gamut mapping where a 3D LUT is required.

HDR DISPLAY PIPELINE

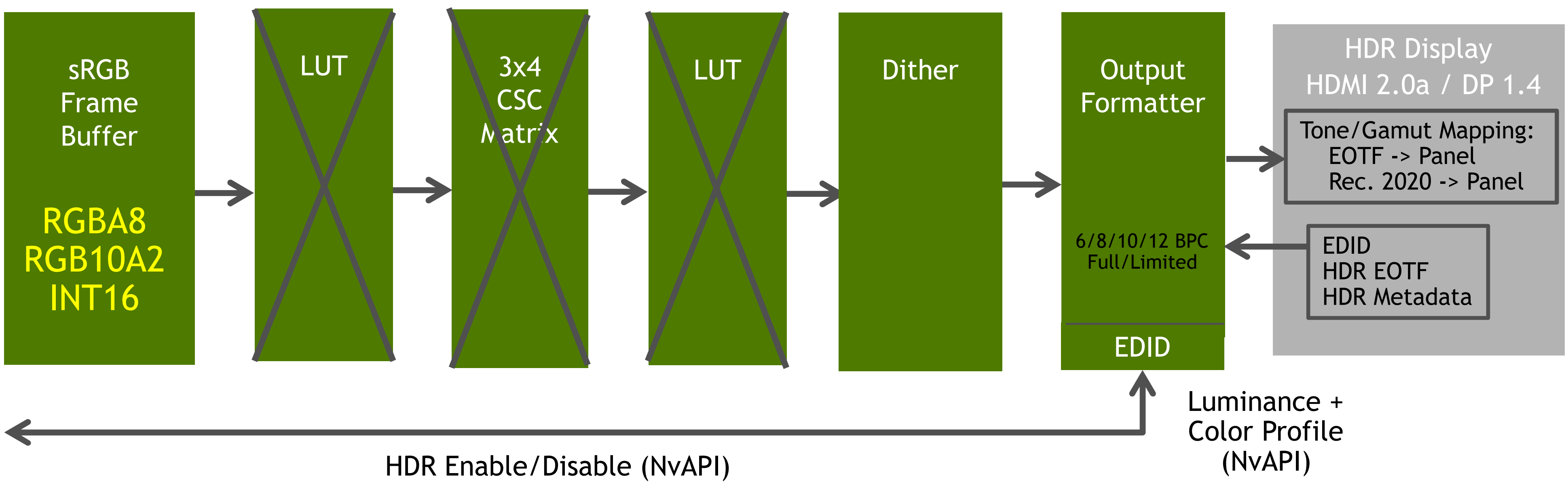
Case 1: Application Renders Linear Content



TT1

HDR DISPLAY PIPELINE (WIN 7 ONLY)

Case 2: Application Makes Display Ready Content



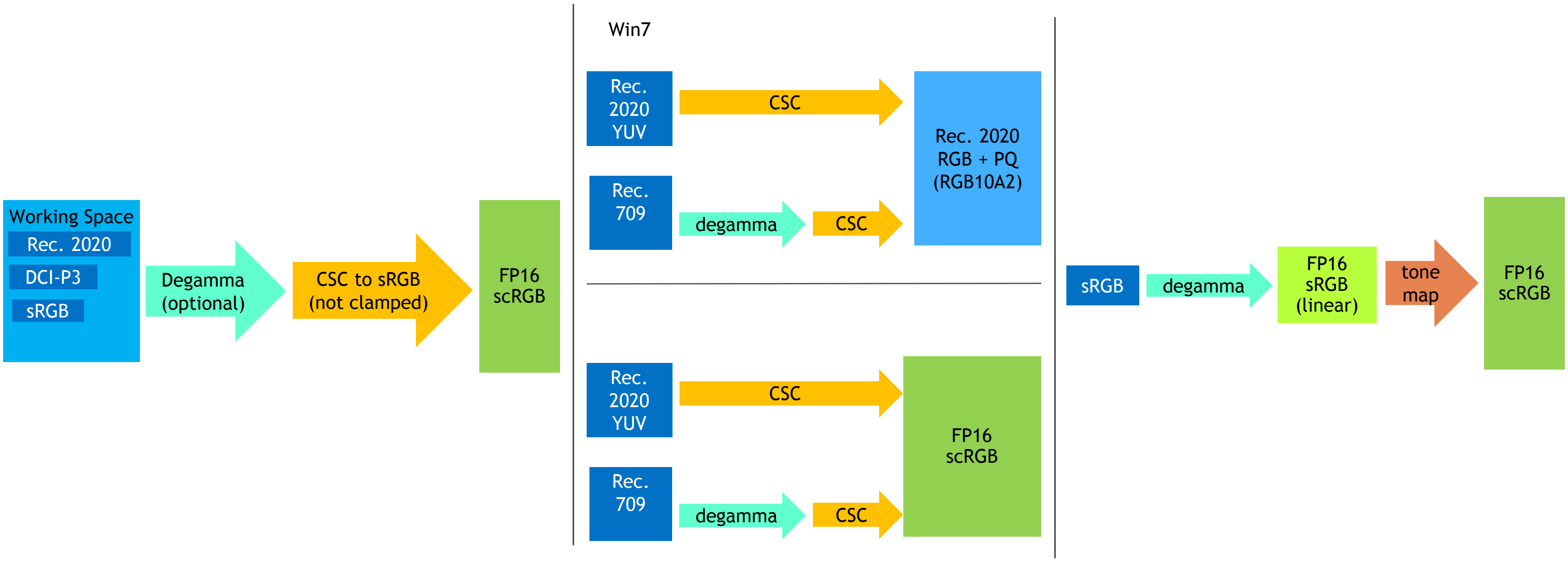
Slide 44

TT1	Where does RGB to YUV conversion happen for YUV displays -- DIsplay Engine? Thomas True, 4/14/2017

PROGRAMMING FOR HDR

HDR APPLICATION PATHS

Color Management Aware | Working With Existing HDR Content | sRGB/Not Color Managed



HDR RENDERING PIPELINE

Practical Path to Utilizing Current HDR Displays

1) Create content with sRGB primaries as done today for SDR.
2) Render high-quality HDR using physically-based shading.
3) Post process in the scene referred space
4) Apply color grading to the rendered scene referred image
5) Tone map with a filmic ACES-derived tone mapper
6) Keep backbuffer in FP16 scRGB
7) Composite 8-bit sRGB referenced UI as normal

DISPLAYING HDR ON WINDOWS

A Win7/Win10 Pre-RS2 Quick Start Guide

Create backbuffer as R16G16B16A16_FLOAT/FP16 - Ensures enough color precision

- DirectX: Create **DXGI_FORMAT_R16G16B16A16_FLOAT** swap chain
- OpenGL: Specify `WGL_PIXEL_TYPE_ARB = `**`WGL_TYPE_RGBA_FLOAT_ARB`**
  with color depth 16 (`WGL_RED_BITS_ARB = `**`16`**`,`
  `WGL_GREEN_BITS_ARB = `**`16`**`, WGL_BLUE_BITS_ARB = `**`16`**)

Make window fullscreen exclusive - Prevents OS compositor from destroying data

Query HDR capability from NVAPI

Call NVAPI to send HDR metadata and enable HDR

Output linear tone mapped scene to FP16 scRGB backbuffer in scRGB colorspace

DISPLAYING HDR ON WINDOWS

A Win10 RS2 Quick Start Guide

HDR mode enabled when HDR10 display detected.

Create backbuffer as R16G16B16A16_FLOAT/FP16 - Ensures enough color precision

DirectX: Create **DXGI_FORMAT_R16G16B16A16_FLOAT** swap chain

OpenGL: Specify `WGL_PIXEL_TYPE_ARB = ` **`WGL_TYPE_RGBA_FLOAT_ARB`**
with color depth 16 (`WGL_RED_BITS_ARB = ` **`16`**`,`
`WGL_GREEN_BITS_ARB = ` **`16`**`, WGL_BLUE_BITS_ARB = ` **`16`**)

Create HDR Window

Output linear tone mapped scene to FP16 scRGB backbuffer in scRGB colorspace

DISPLAYING HDR ON WINDOWS

Use NVAPI to Enumerate GPUs and Connected Displays

```
// Enumerate GPUs and connected displays

NvPhysicalGpuHandle *nvGPUHandle = (NvPhysicalGpuHandle *)calloc(NVAPI_MAX_PHYSICAL_GPUS,
                                                                 sizeof(NvPhysicalGpuHandle));;
NvU32 nvGPUCount;
NvU32 *nvConnectedDisplayIdCount = (NvU32 *)calloc(NVAPI_MAX_DISPLAYS, sizeof(NvU32));

NV_GPU_DISPLAYIDS **nvConnectedDisplayIds = (NV_GPU_DISPLAYIDS **)calloc(NVAPI_MAX_PHYSICAL_GPUS,
                                                                        sizeof(NV_GPU_DISPLAYIDS));

if (EnumerateGPUsAndDisplays(nvGPUHandle, &nvGPUCount, nvConnectedDisplayIds,
                             nvConnectedDisplayIdCount) != NVAPI_OK)
{
    MessageBox(NULL, TEXT("GPU and Display Enumeration Failed."), applicationTitle,
               MB_OK | MB_ICONINFORMATION);
    return 0;
}
```

DISPLAYING HDR ON WINDOWS

Query HDR Capabilities of Each Display from NVAPI

```
// On each GPU, get the HDR capabilities of each active display.
NvU32 gpu = 0;
NvU32 display = 0;
while (gpu < nvGPUCount)
{
      while (display < nvConnectedDisplayIdCount[gpu])
      {
            NV_HDR_CAPABILITIES hdrCapabilities;
            if (NvAPI_Disp_GetHdrCapabilities(display, &hdrCapabilities) != NVAPI_OK)
            {
                  MessageBox(NULL, TEXT("NVAPI GetHdrCapabilities Failed."),
                              applicationTitle, MB_OK | MB_ICONINFORMATION);
                  return 0;
            }

            // If HDR is supported, break
            if (hdrCapabilities.isST2084EotfSupported)
                  break;

            display++;
      }
      gpu++
}
```

DISPLAYING HDR ON WINDOWS

Display Capabilities

```
struct                                  //!< Static Metadata Descriptor Type 1, CEA-861.3, SMPTE ST2086
{
     NvU16    displayPrimary_x0;           //!< x coordinate of red primary ([0x0000-0xC350] = [0.0 - 1.0])
     NvU16    displayPrimary_y0;           //!< y coordinate of red primary ([0x0000-0xC350] = [0.0 - 1.0])

     NvU16    displayPrimary_x1;           //!< x coordinate of green primary ([0x0000-0xC350] = [0.0 - 1.0])
     NvU16    displayPrimary_y1;           //!< y coordinate of green primary ([0x0000-0xC350] = [0.0 - 1.0])

     NvU16    displayPrimary_x2;           //!< x coordinate of blue primary ([0x0000-0xC350] = [0.0 - 1.0])
     NvU16    displayPrimary_y2;           //!< y coordinate of blue primary ([0x0000-0xC350] = [0.0 - 1.0])

     NvU16    displayWhitePoint_x;         //!< x coordinate of white point ([0x0000-0xC350] = [0.0 - 1.0])
     NvU16    displayWhitePoint_y;         //!< y coordinate of white point ([0x0000-0xC350] = [0.0 - 1.0])

     NvU16    desired_content_max_luminance;     //!< Maximum display luminance = desired max luminance of
                                                 //!< HDR content ([0x0001-0xFFFF] = [1.0 - 65535.0] cd/m^2)
     NvU16    desired_content_min_luminance;     //!< Minimum display luminance = desired min luminance of
                                                 //!< HDR content ([0x0001-0xFFFF] = [1.0 - 6.55350] cd/m^2)
     NvU16    desired_content_max_frame_average_luminance; //!< Desired maximum Frame-Average Light Level
                                                           //!< (MaxFALL) of HDR content
                                                           //!< ([0x0001-0xFFFF] = [1.0 - 65535.0] cd/m^2)
    }display_data;
```

DISPLAYING HDR ON WINDOWS

Call NVAPI To Send HDR Metadata and Enable HDR

```
// If HDR is supported, enable it
if (hdrCapabilities.isST2084EotfSupported)
{
     NV_HDR_COLOR_DATA hdrColorData = {};
     memset(&hdrColorData, 0, sizeof(hdrColorData));

     hdrColorData.version = NV_HDR_COLOR_DATA_VER;
     hdrColorData.cmd = NV_HDR_CMD_SET;
     hdrColorData.static_metadata_descriptor_id = NV_STATIC_METADATA_TYPE_1;
     hdrColorData.hdrMode = NV_HDR_MODE_UHDA;

     if(NvAPI_Disp_HdrColorControl(display, &hdrColorData) != NVAPI_OK)
     {
          MessageBox(NULL, TEXT("NVAPI HdrColorControl Failed."), applicationTitle,
                   MB_OK | MB_ICONINFORMATION);
          return 0;
     }
}
```

DISPLAYING HDR ON WINDOWS

Enabling DolbyVision

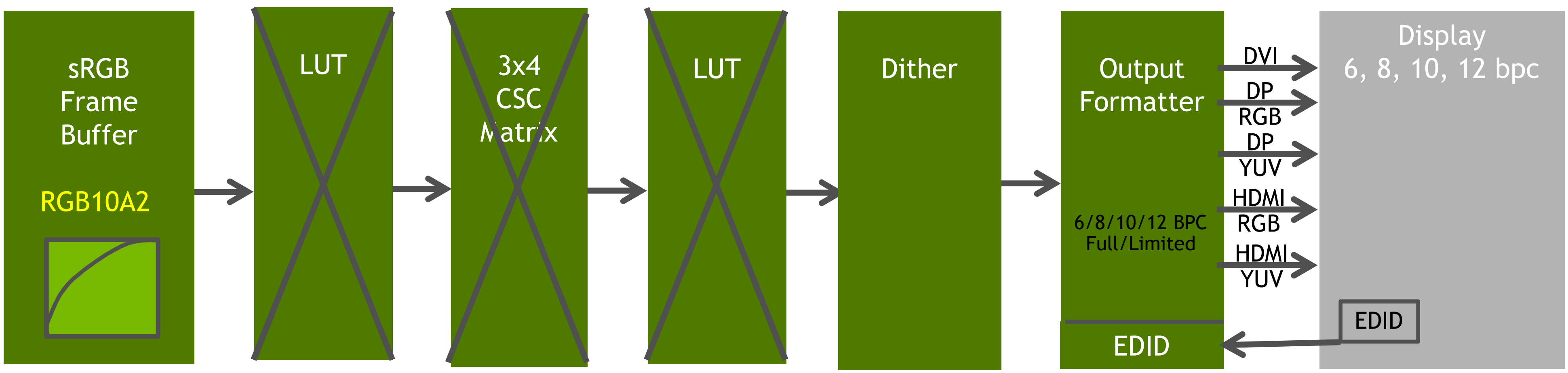
```
// If HDR is supported, enable it
if (hdrCapabilities.isDolbyVisionSupported)
{
      NV_HDR_COLOR_DATA hdrColorData = {};
      memset(&hdrColorData, 0, sizeof(hdrColorData));

      hdrColorData.version = NV_HDR_COLOR_DATA_VER;
      hdrColorData.cmd = NV_HDR_CMD_SET;
      hdrColorData.static_metadata_descriptor_id = NV_STATIC_METADATA_TYPE_1;
      hdrColorData.hdrMode = NV_HDR_MODE_DOLBY_VISION;

      if(NvAPI_Disp_HdrColorControl(display, &hdrColorData) != NVAPI_OK)
      {
            MessageBox(NULL, TEXT("NVAPI HdrColorControl Failed."), applicationTitle,
                      MB_OK | MB_ICONINFORMATION);
            return 0;
      }
}
```

DISPLAYING DOLBYVISION ON WINDOWS

DolbyVision Static Metadata

```
struct
{
      NvU32 VSVDB_version             : 3;               //!< Version of Vendor Data block,Version 0: 25 bytes  Version 1: 14 bytes
      NvU32 dm_version                : 8;               //!< Upper Nibble represents major version of Display Management(DM)
                                                         //!< while lower represents minor version of DM
      NvU32 supports_2160p60hz        : 1;               //!< If set sink is capable of 4kx2k @ 60hz
      NvU32 supports_YUV422_12bit     : 1;               //!< If set, sink is capable of YUV422-12 bit
      NvU32 supports_global_dimming   : 1;               //!< Indicates if sink supports global dimming
      NvU32 colorimetry               : 1;               //!< If set indicates sink supports DCI P3 colorimetry, REc709 otherwise
      NvU32 reserved                  : 17;              //!< Should be set to zero
                                                         //!< All values below are encoded use
                                                         //!< DolbyVisionHDMITransmission Specification document to decode
      NvU16 target_min_luminance;                        //!< Represents min luminance level of Sink
      NvU16 target_max_luminance;                        //!< Represents max luminance level of sink
      NvU16 cc_red_x;                                    //!< Red primary chromaticity coordinate x
      NvU16 cc_red_y;                                    //!< Red primary chromaticity coordinate y
      NvU16 cc_green_x;                                  //!< Green primary chromaticity coordinate x
      NvU16 cc_green_y;                                  //!< Green primary chromaticity coordinate Y
      NvU16 cc_blue_x;                                   //!< Blue primary chromaticity coordinate x
      NvU16 cc_blue_y;                                   //!< Blue primary chromaticity coordinate y
      NvU16 cc_white_x;                                  //!< White primary chromaticity coordinate x
      NvU16 cc_white_y;                                  //!< White primary chromaticity coordinate y
}dv_static_metadata;
```

DISPLAYING HDR ON LINUX



HDR BEST PRACTICES

HDR BEST PRACTICES

Physically-Based Rendering

Makes light interactions more correct / plausible.

- Results in proper highlights, not just a hack that looks good in SDR

Creates values on a scale consistent with the real world

- [0,1] brightness level doesn’t make sense in an HDR world
- Will need to make compromises (FP16 won’t represent the brightest sun)

HDR BEST PRACTICES

Colorspace

Keep using sRGB primaries.

Keeps consistency with the present art pipeline

- No surprises for artists
- No gamut mapping problems on SDR displays

Will still reap the benefits of brightness and brighter saturated colors

Starting point. Plan to be more aggressive in the future.

HDR BEST PRACTICES

Gamut Remapping

Stretches rendering done with sRGB primaries to more extreme ones.

- Produces richer / more saturated colors
- May work OK for some applications, not so much for others

Will present challenges with existing artwork

- Unnatural skin tones
- Hue shifting
- Memory color

HDR BEST PRACTICES

Luminous Effects

Make things that glow, glow at a level consistent with the light source

- Emissive level and light source should be correlated

Looks odd when a specular high light outshines a light source

HDR BEST PRACTICES

Scene Referred Post-FX

Perform operations that require linear lighting prior to tone mapping.

- Bloom
- Motion blur
- Depth of field

Operating scene-referred maintains consistency

- Same operation for HDR and SDR

HDR BEST PRACTICES

Color Grading

Perform in scene-referred space

- Makes operations consistent SDR/HDR
- Avoids tone mapping “fix-ups”

HDR BEST PRACTICES

Luminance-Aware Tone Mapping

Many operators designed to work within a generic [0-1] space.

- How bright is 1.0?
- Scaling to a 1000 nit display would display asphalt at 100 nits

Need an operator that understands the output luminance

- Middle gray stays at a reasonable level
- Colors only compressed where they need to be

Use ACES or Drago

HDR BEST PRACTICES

UI Compositing

UI typically authored in straight RGB

- Need to composite properly into the color space of the HDR scene

Using scRGB backbuffer provides simple solution

- Same sRGB primaries with (1, 1, 1) as the brightness level for white
- Blending just works.

HDR BEST PRACTICES

UI Blending Challenges

UI may look dimmer / duller than intended

- Due to adaptation of the eye to the brighter colors of the HDR display
- Scale up the UI luminance to counteract this

Transparent elements may suffer glow throw effects

- Example: 1000 nit highlight behind a transparent dialog with white text
- Solution: Clip / Apply simple SDR tone map to scene elements underneath

HDR SDK

HDR Display Sample Code

Simple app demonstrating ACES tone mappers + HDR display

Allows setting HDR metadata to enable HDR on UHD TVs

Offers standard ACES tone mappers + customized ACES tone mappers

Provides EXR and HDR file format loading to visualize HDR data

Offers exposure scaling and range ‘enhancement’ tools

http://developer.nvidia.com/high-dynamic-range-display-develpment

CONCLUSION

KEY TAKE AWAYS

- HDR displays are here today with more available soon.
- HDR will help your application tell a better story.
- Three types of HDR applications
- Importance of supporting an scRGB frame buffer
- Luminance aware tone mapping to support HDR as well as SDR
- Leverage NvAPI to determine HDR display capabilities

Q & A