ASTC
(Adaptive Scalable Texture Compression)

ARM

Stacy Smith
Senior Software Engineer, ARM

ARM Game Developer Day - London
03/12/2015

Outline

- Introduction of texture compression and block-based algorithms
- Previous texture compression formats vs ASTC
- Use cases for ASTC 2D and 3D textures

Texture Compression

- Modern graphical applications use lots of textures…

Texture Compression

…but textures are big

- At 24 bits/pixel, multi-megabytes

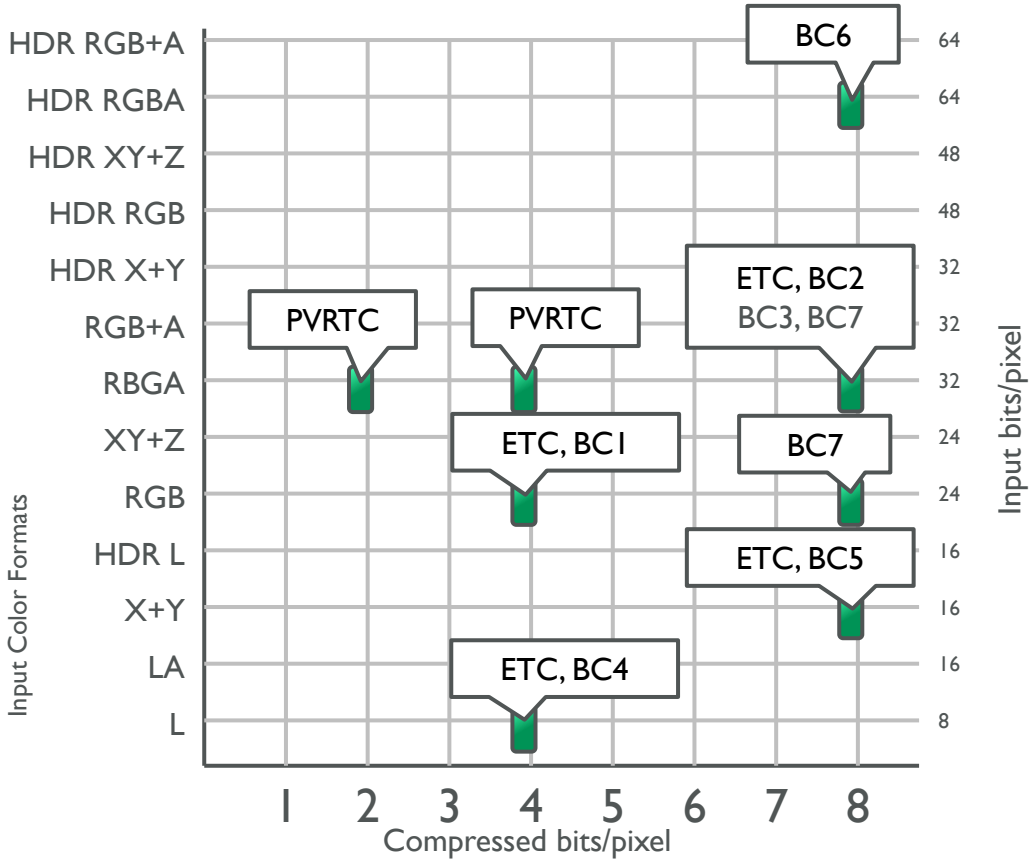
- This is a problem
  - Big downloads
  - Big memory footprint
  - Big memory bandwidth
  - Big power drain

- So we have to compress them

- Hardware decompression saves memory and bandwidth

Texture Compression

- Different codecs support different footprints and bitrates
- They also support different color encodings (HDR / sRGB / alpha)
- Hardware limits access to proprietary codecs



ARM

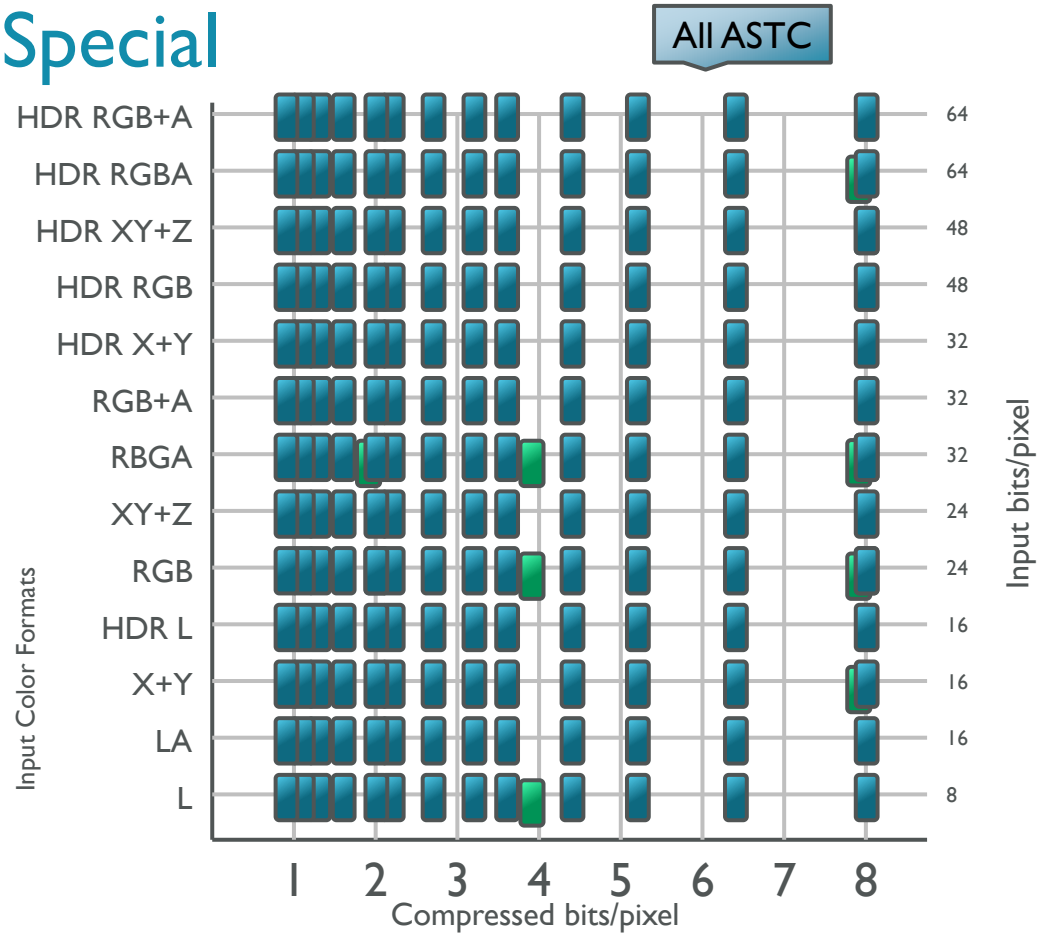
What Makes ASTC Special

- Wide range of bitrates
- Wide range of formats

- Handles sRGB
- Handles HDR

- 3D Textures

- Non Proprietary



ARM

ASTC: 2D Texture Support

Use case: porting Seemore demo to ASTC

Compared to ETC2:

- Using ASTC compression with 5x5 block size gave us ~24% of memory reduction whilst maintaining the same quality level (measured using PSNR)
- Using ASTC 6x6 gave us a ~45% memory reduction with negligible quality loss
- As a consequence, the memory read bandwidth required for texturing reduced; improving the performance.

Textures Total Size MB	
Uncompressed	263.00
ETC2	93.10
ASTC 5x5	71.00
ASTC 6x6	49.40
Memory Read Bandwidth in MB/s	
Uncompressed	1476.87
ETC2	749.59
ASTC 5x5	664.9
ASTC 6x6	595.02
Energy consumption DDR2 mJ per frame	
Uncompressed	3.32
ETC2	1.687
ASTC 5x5	1.496
ASTC 6x6	1.339

ASTC: 3D Texture Support

Use case: ASTC 3D texture for particles collision

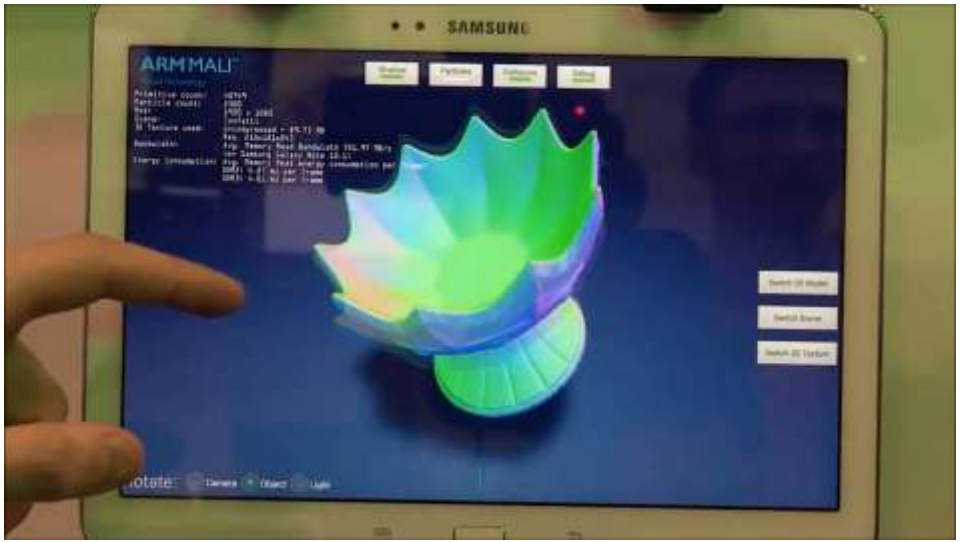
- Various block sizes to choose from
- Each compressed block will still occupy 128 bits.
- HDR support allows us to store 16Bit half-floats

Block Dimension	Bit Rate (bits per texel)
3x3x3	4.74
4x3x3	3.56
4x4x3	2.67
4x4x4	2.00
5x4x4	1.60
5x5x4	1.28
5x5x5	1.02
6x5x5	0.85
6x6x5	0.71
6x6x6	0.59

ASTC: 3D Texture Support

Use case: ASTC 3D texture for particles collision

- Since the simulation is run in the vertex shader we need to provide information about the environment for collisions
- Voxelizing the environment allows us to save it as a 3D texture
- But the data generated is huge….so lets compress it with ASTC 3D

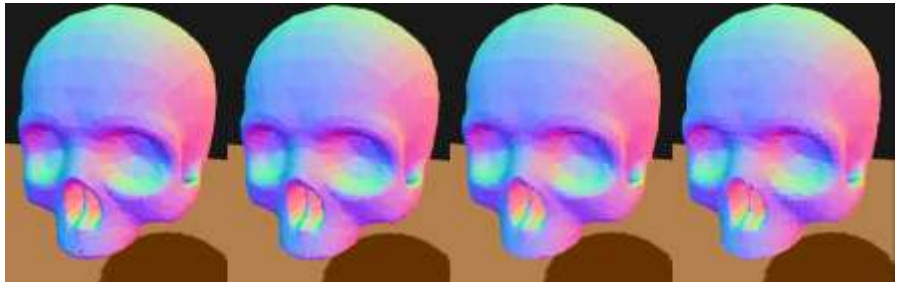


ARM

ASTC: 3D Texture Support

Use case:

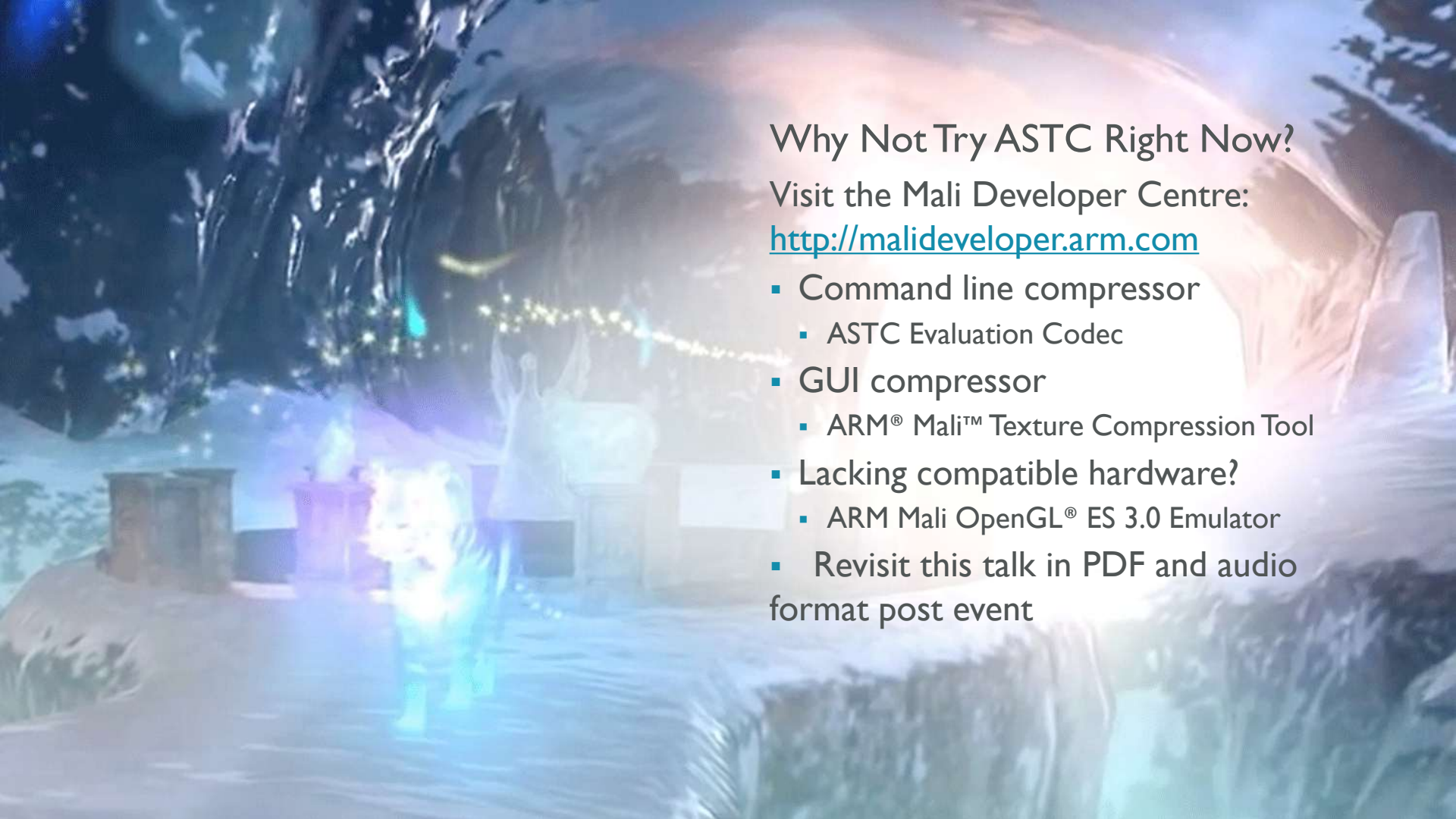
ASTC 3D texture for particles collision



- ~90% memory reduction
- ~38% memory bandwidth reduction

	Skull
Texture Resolution	180x255x255
Texture Size MB	
Uncompressed	82.62
ASTC 3x3x3	6.12
ASTC 4x4x4	2.63
ASTC 5x5x5	1.32
Memory Read Bandwidth in MB/s	
Uncompressed	752.18
ASTC 3x3x3	285.78
ASTC 4x4x4	179.43
ASTC 5x5x5	167.90
Energy consumption per frame DDR2 mJ per frame	
Uncompressed	5.08
ASTC 3x3x3	1.93
ASTC 4x4x4	1.21
ASTC 5x5x5	1.13
Energy consumption per frame DDR3 mJ per frame	
Uncompressed	4.17
ASTC 3x3x3	1.59
ASTC 4x4x4	1.00
ASTC 5x5x5	0.93

ARM



Why Not Try ASTC Right Now?

Visit the Mali Developer Centre:
http://malideveloper.arm.com

- Command line compressor
  - ASTC Evaluation Codec
- GUI compressor
  - ARM® Mali™ Texture Compression Tool
- Lacking compatible hardware?
  - ARM Mali OpenGL® ES 3.0 Emulator
- Revisit this talk in PDF and audio format post event

Thank you

ARM

Questions