



# ESPN Emerging Technology



## GTC Session S3487

ESPN Emerging Technology's use of  
NVIDIA'S GPU Solutions for High  
Resolution Imagery







# ESPN Emerging Technology

## Overview

- Who we are / What we do
- Software Architecture built around NVIDIA GPUs
- ESPN Uses of this SW Architecture
- Q & A





# ESPN Emerging Technology

Who we are / What we do

## ESPN Emerging Technology (ET)

ET develops the ideas and applications that bring the fan closer to the game with new, innovative production enhancements.

- Huck-O-Meter, HRD Ball Track, Snap Zoom, Ref Mics, Sky Cam, Ultra-Mo, Player Tracking, the 1st & Ten Line, K-Zone, the Emmy-winning EA Virtual Playbook and much, much more.



# ESPN Emerging Technology

Who we are / What we do

ESPN Wide World of Sports Innovation Lab in Orlando

- Designated testing ground for new innovations
- Over 300 sporting events occur at this facility each year



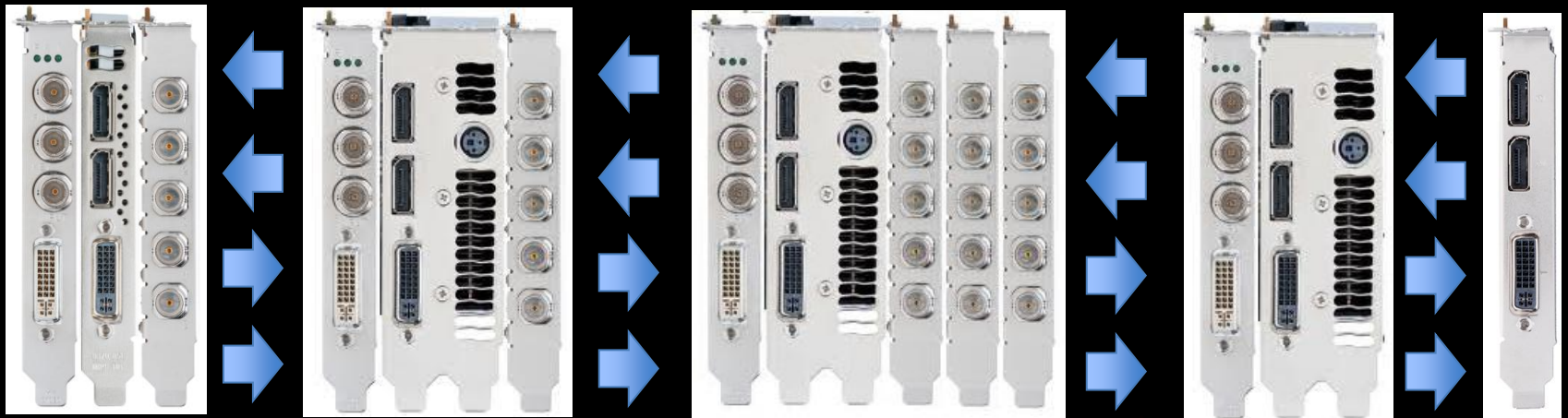
**ESPN**



# ESPN Emerging Technology

Software Architecture built around NVIDIA GPUs

- Each GPU in the system is classified as either an Input, Output, Input / Output, or a Compute Engine
- All GPUs have peer-to-peer access via CUDA
- Multiple input cards and output cards may be assigned to a GPU



# ESPN Emerging Technology

Software Architecture built around NVIDIA GPUs

- Hardware abstraction layer allows video I/O hardware from several manufacturers via gpuDirect



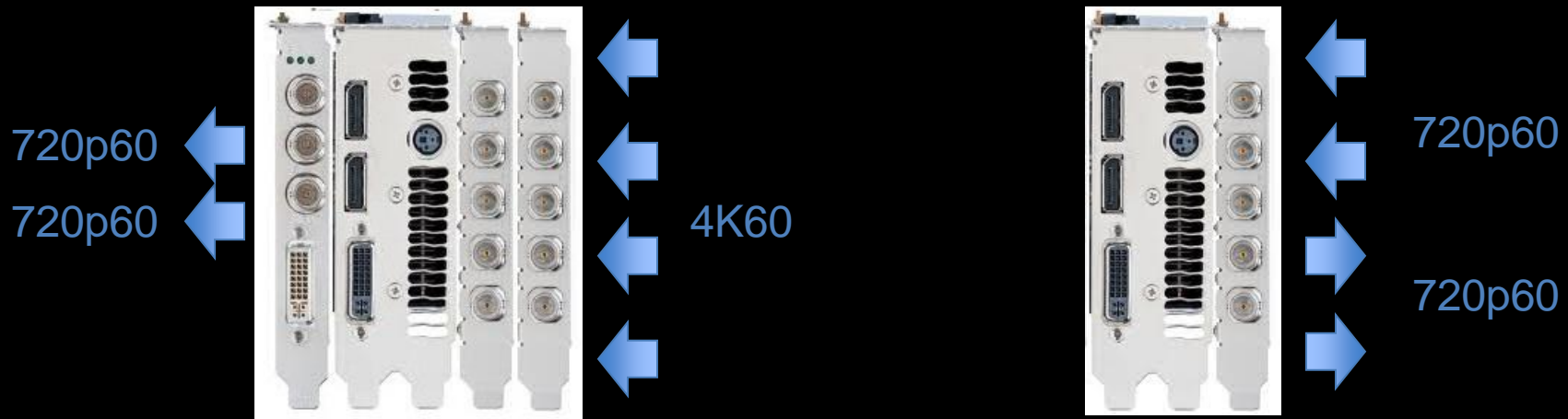
- Supports Quadro and Tesla GPUs (Fermi and Kepler)



# ESPN Emerging Technology

Software Architecture built around NVIDIA GPUs

- Supports use of Cubix GPU Xpander
- Each GPU pipeline can handle unique video formats for input vs. output



**ESPN**



# ESPN Emerging Technology

Video Capture for Advanced Technologies



**ESPN**



# ESPN Emerging Technology

## Low Cost Video Effects



# ESPN Emerging Technology

4k to 720p workflows

4K Input



Zoomed 720p



**ESPN**



# ESPN Emerging Technology

4k to 720p workflows

4K Input



Virtual Cameras



**ESPN**

# ESPN Emerging Technology



Questions?

