Introduction

- IT industry experiences an increasing growth for display surfaces with high resolution
- Use cases for such surfaces include satellite and map data, x-ray and microscope images, multimedia, CCTV, etc.
- Existing solutions are not scalable, do not offer hardware abstraction, suffer from wiring limitations

Currently Popular Monitor Wall Architectures

- OS can natively manage the displays
  - Pro: OS can manage the displays
  - Con: Power consumption, supported monitor count limited by the output count of the GPUs and expansion slots for the GPUs on the motherboard, deployment is limited by wiring

- Display wall architecture where each output of the GPU maps to a tile on the display wall
  - Pro: OS can natively manage the displays
  - Con: Power consumption, supported monitor count limited by the output count of the GPUs and expansion slots for the GPUs on the motherboard, deployment is limited by wiring

- Display wall architecture where each output of the GPU is split/upscaled among the tiles on the display wall
  - Pro: Software complexity is reduced since it does not have to be multiple monitor aware
  - Con: Small resolution and DPI, visualization is not displayed in its native resolution
  - Con: Expensive

Proposed Virtual Machine Based Monitor Wall Architecture

- The proposed display wall architecture where each output of a virtual GPU maps to a tile on the display wall and is transmitted as an H.264 stream over LAN
- No Direct3D, OpenGL support
- Lossy compression
- No Direct3D, OpenGL support

Why NVENC?

- The current experiments show that this architecture is very feasible for non FPS intensive use cases where the display wall can be driven by a single physical GPU
- The total resolution provided by this architecture even using the currently available compression technology greatly exceeds the resolutions of existing solutions, it would be expected for the resolution to grow in the future
- The architecture itself scales very good, it is limited mainly by OS support for multiple monitors (this can be overcome by simulating a single high resolution display in the virtual machine that spans the whole resolution of the physical wall) and the possibility to stack multiple GPU’s in the host system
- Future work should focus on the ability to virtualize OpenGL and Direct3D to remove the advantages of non-virtualized architectures

Conclusions