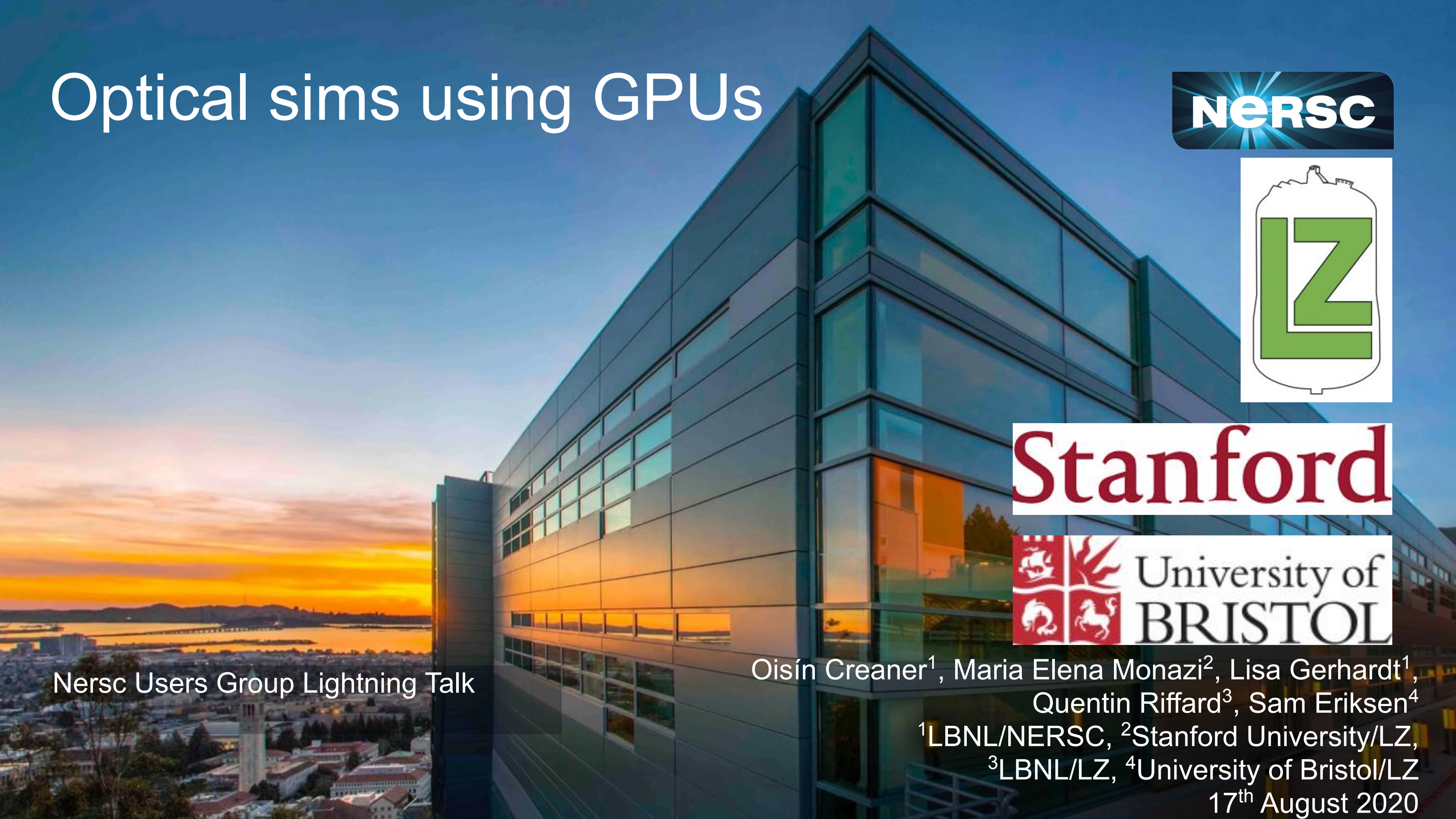
Optical sims using GPUs



Nersc Users Group Lightning Talk

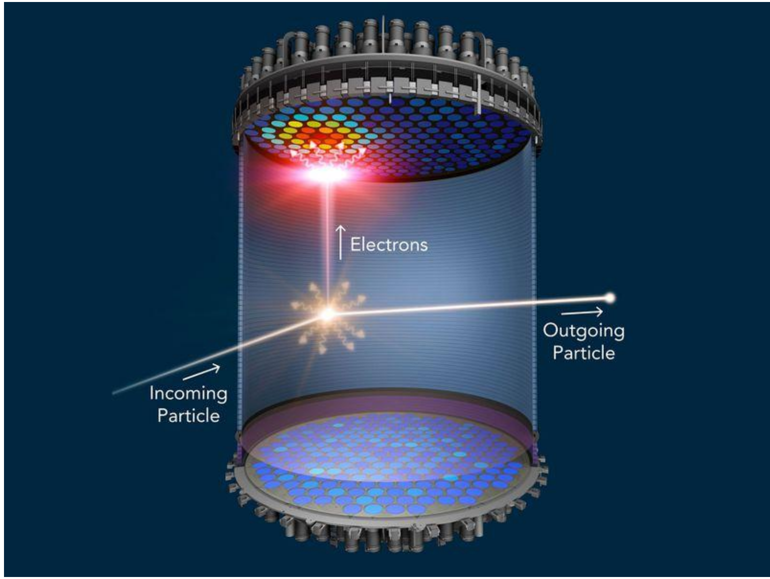
Oisín Creaner[1], Maria Elena Monazi[2], Lisa Gerhardt[1], Quentin Riffard[3], Sam Eriksen[4]

[1]LBNL/NERSC, [2]Stanford University/LZ, [3]LBNL/LZ, [4]University of Bristol/LZ

17th August 2020

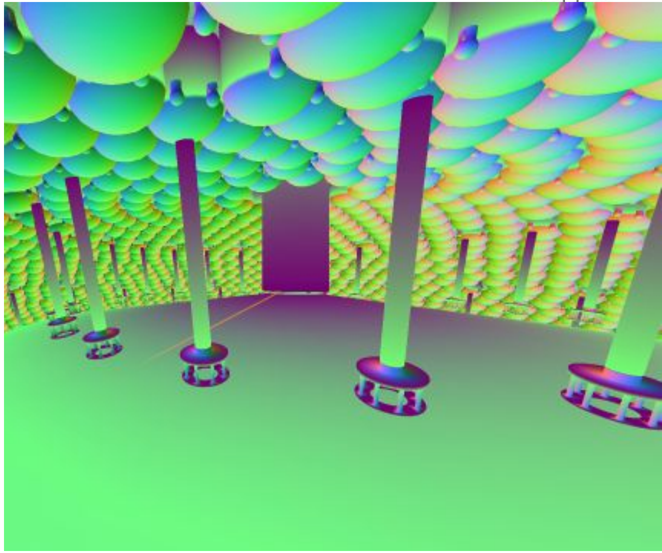
LZ

- LUX-ZEPPLIN (Large Underground Xenon ZonEd Proportional scintillation in LIquid Noble gases)
- Next-generation Dark Matter detector
- Particle interactions with the Xenon detection medium
- Compare with simulated interactions.
- Photon propagation: time- and resource-intense



Motivation to GPUs

- Replace existing CPU module with GPU solution
  - Photon propagation is simulation bottleneck
  - Faster simulations
  - More frequent simulations
  - Better use of available hardware
- Opticks: A GPU Accelerated Optical Photon Simulation using NVIDIA OptiX



Blythe, S (2019) Renders of the chimney region of JUNO detector as an Opticks Geometry

LZ at NERSC

- LZ is major NERSC user

- Simulations: CPU based
  Data storage
  Data analysis:

- Mock Data Challenge
  - Science Run 1 starting soon!

- Existing Optics implementation Local/Cloud
  - Move to CoriGPU
  - Prepare for Perlmutter

NERSC, 2019

Containerisation: Docker and Shifter

- Shifter:
  - Security requirement to run on CoriGPU (and Perlmutter)
  - Build a docker image, export to Shifter to run
  - READ ONLY and NO ROOT ACCESS
- Docker:
  - Write Dockerfile with all commands required to build
  - Create image with all depencencies and specific versions
  - Should be very portable: solution should continue to work when upgraded

Challenges in creating an image

- Many undocumented assumptions
- Hard-coded or semi-hard-coded (e.g. $HOME) paths
- Hardware awareness of compiled libraries
- Security restrictions on writing
- Unexpected write commands buried deep in layers
- NVidia Runtime mismatch with drivers
- Specific version and/or commit dependencies

Solutions

- Containerisation & Portability
    - Versions locked in when image created
- Fork development from stable point
- Identify assumptions and hard-coding and make soft links
- Mount data in writable partitions (e.g. $SCRATCH)
- Hardware-aware compilation has to happen on the correct host

Optical sims using GPUs

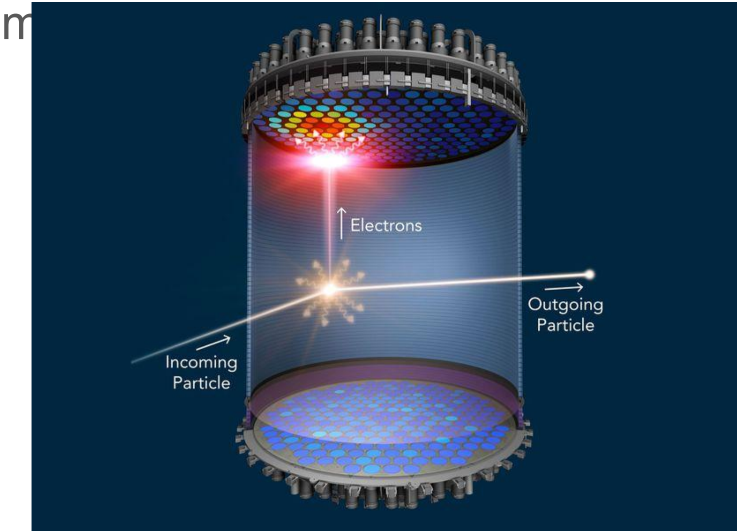
Oisín Creaner, Maria Elena Monazi, Lisa Gerhardt, Quentin Riffard, Sam Eriksen

LZ

LUX-ZEPPLIN (Large Underground Xenon ZonEd Proportional scintillation in LIquid Noble gases)

Next-generation Dark Matter detector

Particle interactions with the Xenon detection medium



Compare with simulated interactions.

Photon propagation: time- and resource-intense

Motivation to GPUs

Replace existing CPU module with GPU solution

- Photon propagation is simulation bottleneck
- Faster simulations
- More frequent simulations
- Better use of available hardware

Opticks: A GPU Accelerated Optical Photon Simulation using NVIDIA OptiX

LZ at NERSC

LZ is major NERSC user

- Simulations: CPU based
- Data storage
- Data analysis:
  - Mock Data Challenge
  - Science Run 1 starting soon!

Existing Optics implementation Local/Cloud

- Move to CoriGPU
- Prepare for Perlmutter

Containerisation: Docker and Shifter

Shifter:

- Security requirement to run on CoriGPU (and Perlmutter)
- Build a docker image, export to Shifter to run
- READ ONLY and NO ROOT ACCESS

Docker:

- Write Dockerfile with all commands required to build
- Create image with all depencencies and specific versions
- Should be very portable: solution should continue to work when upgraded

Challenges in creating an image

Many undocumented assumptions

Hard-coded or semi-hard-coded (e.g. $HOME) paths

Hardware awareness of compiled libraries

Security restrictions on writing

Unexpected write commands buried deep in layers

NVidia Runtime mismatch with drivers

Specific version and/or commit dependencies

Solutions

Containerisation & Portability

Versions locked in when image created

Fork development from stable point

Identify assumptions and hard-coding and make soft links

Mount data in writable partitions (e.g. $SCRATCH)

Hardware-aware compilation has to happen on the correct host

Image References

Blythe, S, 2019 Opticks : GPU Optical Photon Simulation for Particle Physics using NVIDIA® OptiX^{TM}. EPJ Web of Conferences 214, 02027

NERSC 2019, Cori Supercomputer