Intel oneAPI Rendering Toolkit

OSPRay Library

Miroslaw Pawlowski
Agenda

- Introduction
- ospExample application demo
- Building OSPRay
- ospTutorial code walkthrough
- ospExample code walkthrough
- Distributed rendering with MPI
- Where to learn more?
- Q&A
Introduction

- Intel OSPRay is an open source, scalable, and portable ray tracing engine for high-performance, high-fidelity visualization on Intel Architecture CPUs.
- It is part of the Intel oneAPI Rendering Toolkit
- Currently supports Linux, macOS, and Windows
- Internally builds on top of Intel Embree and Open VKL
- Uses code vectorization with Intel ISPC (Implicit SPMD Program Compiler), supports Intel SSE4, AVX, AVX2, and AVX-512
- Released under the permissive Apache 2.0 license
ospExample application demo
Building OSPRay

- Building is not always necessary, Linux, macOS and Windows binaries are available at the OSPRay webpage.
- Source code is placed and maintained at GitHub: [https://github.com/ospray/ospray](https://github.com/ospray/ospray)
  
  ```
  git clone https://github.com/ospray/ospray.git
  ```
- Tools:
  - CMake
  - C++11 compiler (GCC, Clang, MSVC, ICC)
  - ISPC compiler
- Dependencies:
  - Intel Threading Building Blocks (TBB)
  - Intel oneAPI RenderKit common library (rkcommon)
  - Intel oneAPI RenderKit Embree library
  - Intel oneAPI RenderKit Open VKL library
  - Intel oneAPI RenderKit Open Image Denoise (optional)
Building OSPRay, cont.

- There are two build options:
  - Manually download and install all dependencies
    
    ```
    mkdir ospray/build
    cd ospray/build
    ccmake ..
    make -j
    ```

  - Use superbuild:
    
    ```
    mkdir build
    cd build
    cmake [<OSPRAY_SOURCE_LOC>/scripts/superbuild]
    cmake --build .
    ```
ospTutorial code walkthrough
ospExample code walkthrough
Distributed rendering with MPI

- **MPI Offload Rendering**
  - Image parallel rendering only
  - Modifications to the application not required
    
    ```
    mpirun -n <N> ./ospExamples --osp:load-modules=mpi --osp:device=mpiOffload
    ```

- **MPI Distributed Rendering**
  - Image parallel or data parallel rendering
  - Application must load MPI module and create `mpiDistributed` device
  - Special `mpiRaycast` renderer must be used
  - Separate world for each node
OSPRay features summary

Combined geometry and volume rendering

SciVis renderer and path tracer

Huge datasets

OSPRay features:
- CPU (Intel® Xeon® / Core™)
- Advanced materials
- Runs at all scales

Application:
- SciVis renderer and path tracer
- Extensible e.g. SES Geometry, UStuttgart
- Huge datasets 12 billion particles 10TB
- Non-polygonal Geometry
- Implicit isosurfaces
- Streamlines
- Particles / spheres
- Clipping Geometries
- HPC Clusters Cloud Workstations & Clients

Features:
- Extensible
- Advanced materials
- Huge datasets
- Non-polygonal Geometry
- Clipping Geometries
Where to learn more?

- Intel oneAPI Rendering Toolkit:

- Intel OSPRay website:
  https://www.ospray.org

- Intel OSPRay source code:
  https://github.com/ospray/ospray
Q&A