Using Blender for Scientific Visualisation

Sébastien Lemaire - EPCC

ARCHER2 Partners

epcc

THE UNIVERSITY of EDINBURGH

Reusing this material

Scope of presentation

This is an introductory webinar:

- Get an idea of the capabilities given by Blender
- Understand the workflow and requirements
- Know where to look for more information

What is Blender?

- Blender is a free and open-source 3D software
- Used for animated movies, vfx, art, motion graphics
- Includes 3D modelling, texturing, animation tools, simulations (particles, fluid, soft body, hair etc.)
- **Ray tracing rendering**

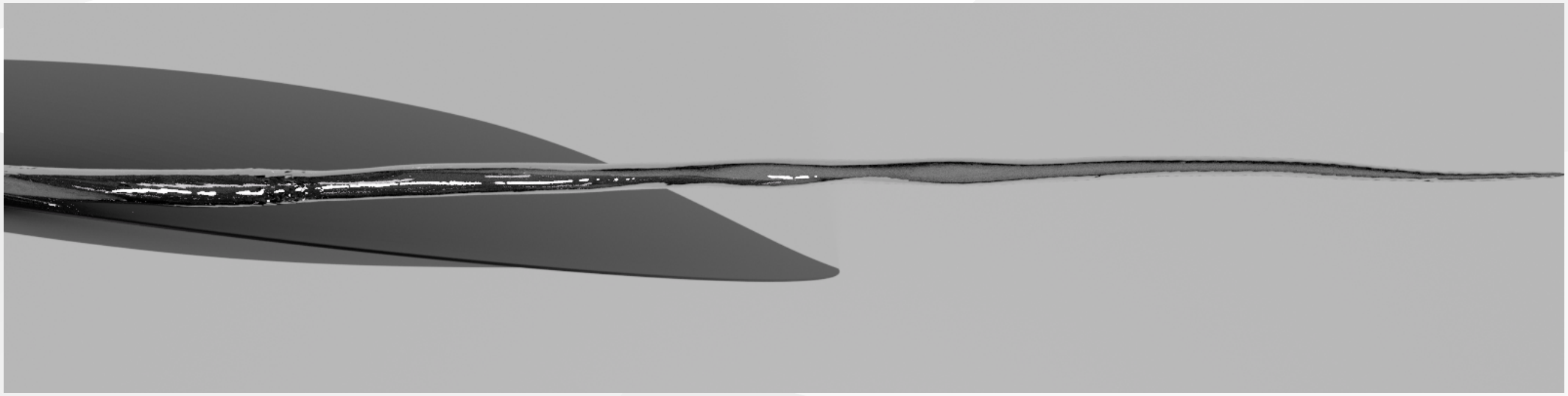
Why use it for scientific visualisation?

- A lot more capable than scientific oriented visualisation tools (e.g. ParaView)
  - Better camera handling, motion
  - Better texturing (water, objects etc.)
  - Easier rendering pipeline
- More engaging visualisation:
  - Outreach
  - Social media
  - Fund applications
  - Papers
- More scientifically accurate rendering (IOR, camera properties etc.)

Basics of visualisation design

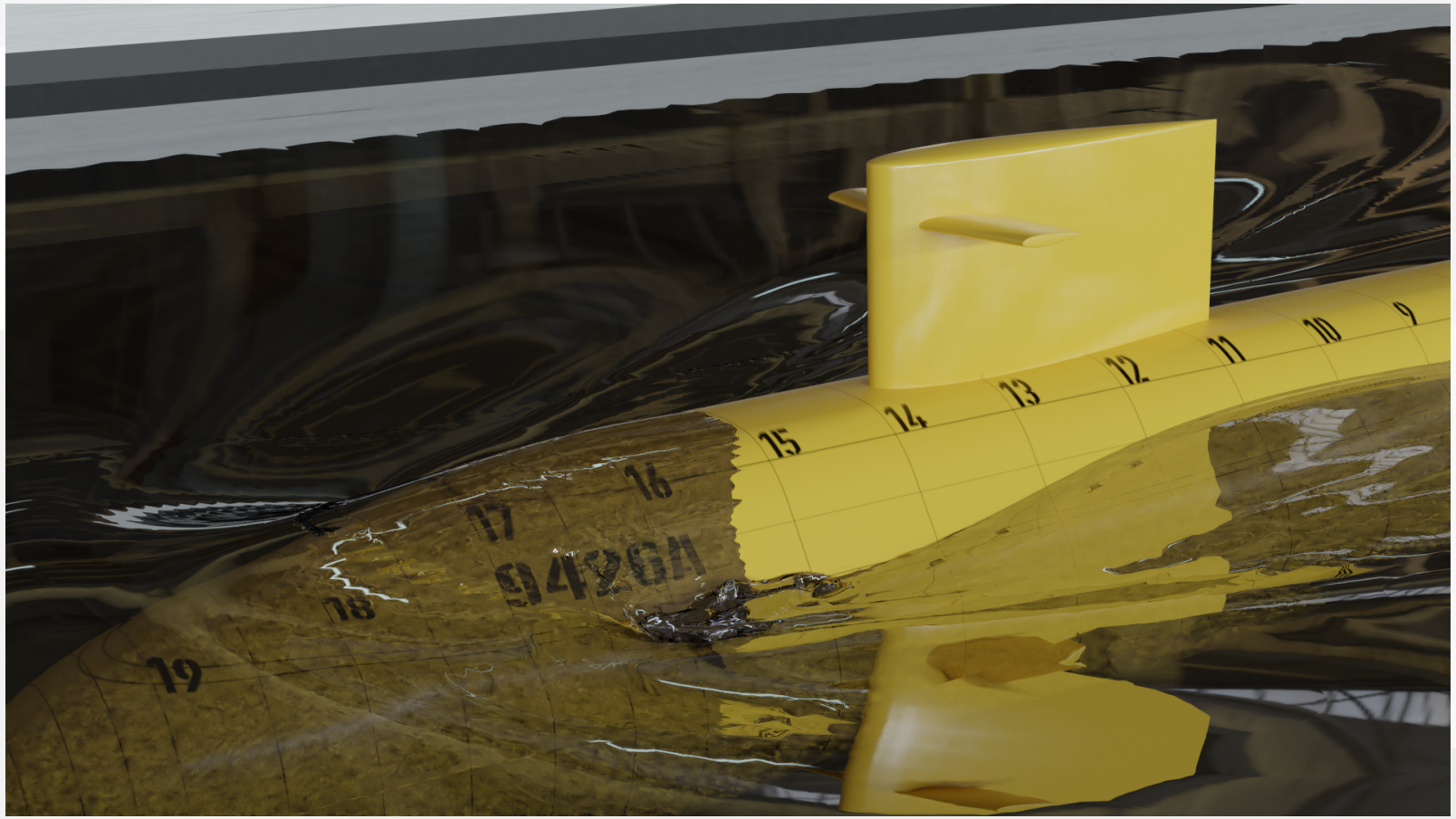
- What is the aim?
  - Explain a concept, show off capabilities, compare with experiments, etc.
- Who is the audience?
  - Vocabulary, Details, Colour scheme, Type
- What is the medium?
  - Background colour, text size, contrast, duration, animation speed etc.

Example



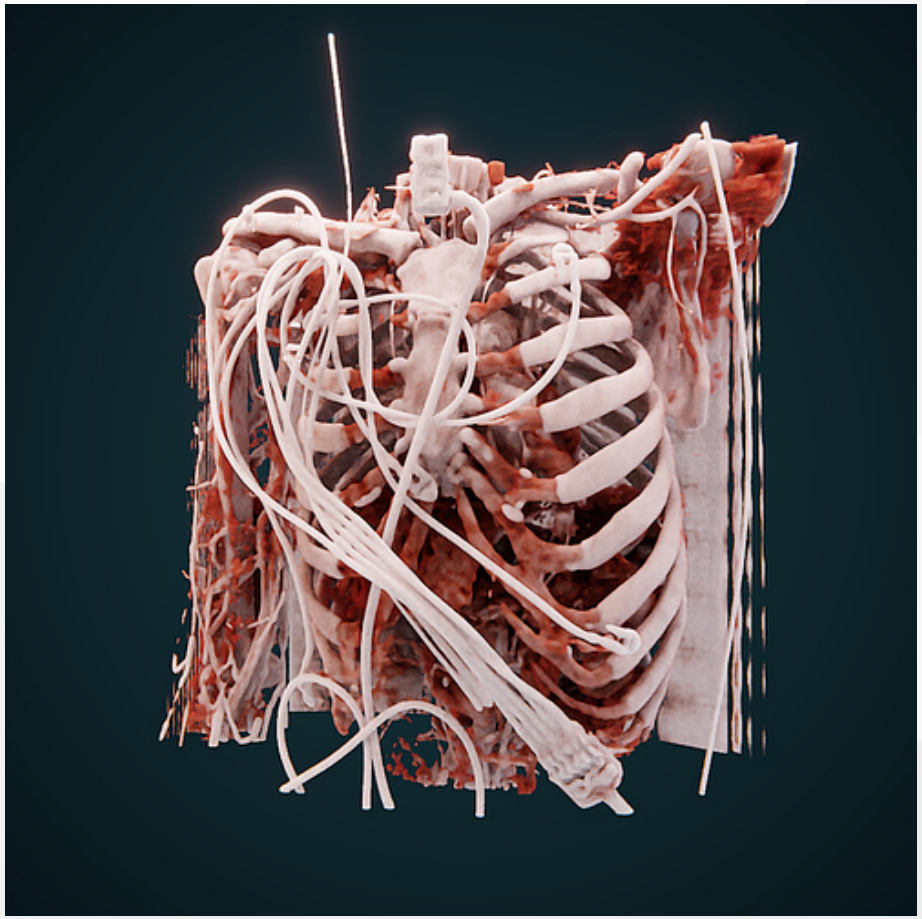
source: [Klapwijk, Maarten (2021).]

Example



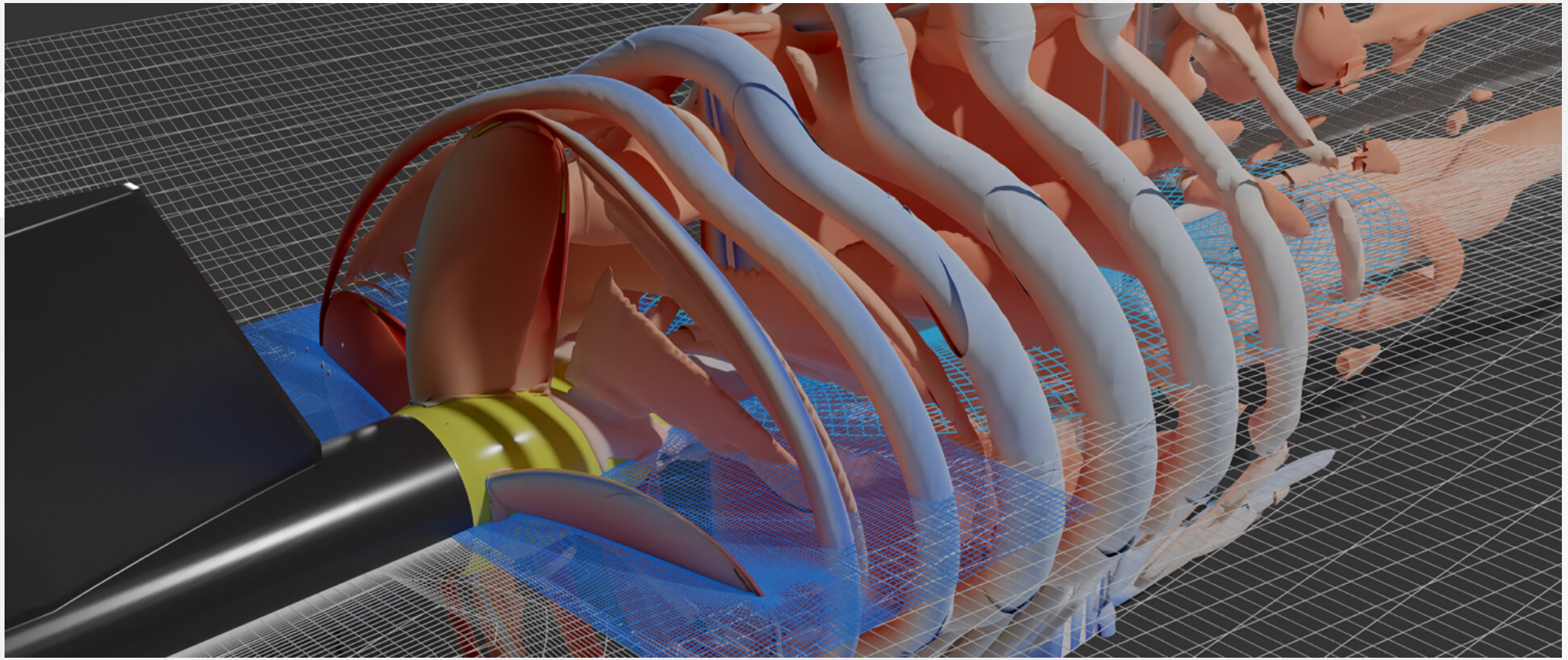
source: [Klapwijk, Maarten.; Lemaire, Sebastien (2021)]

Example



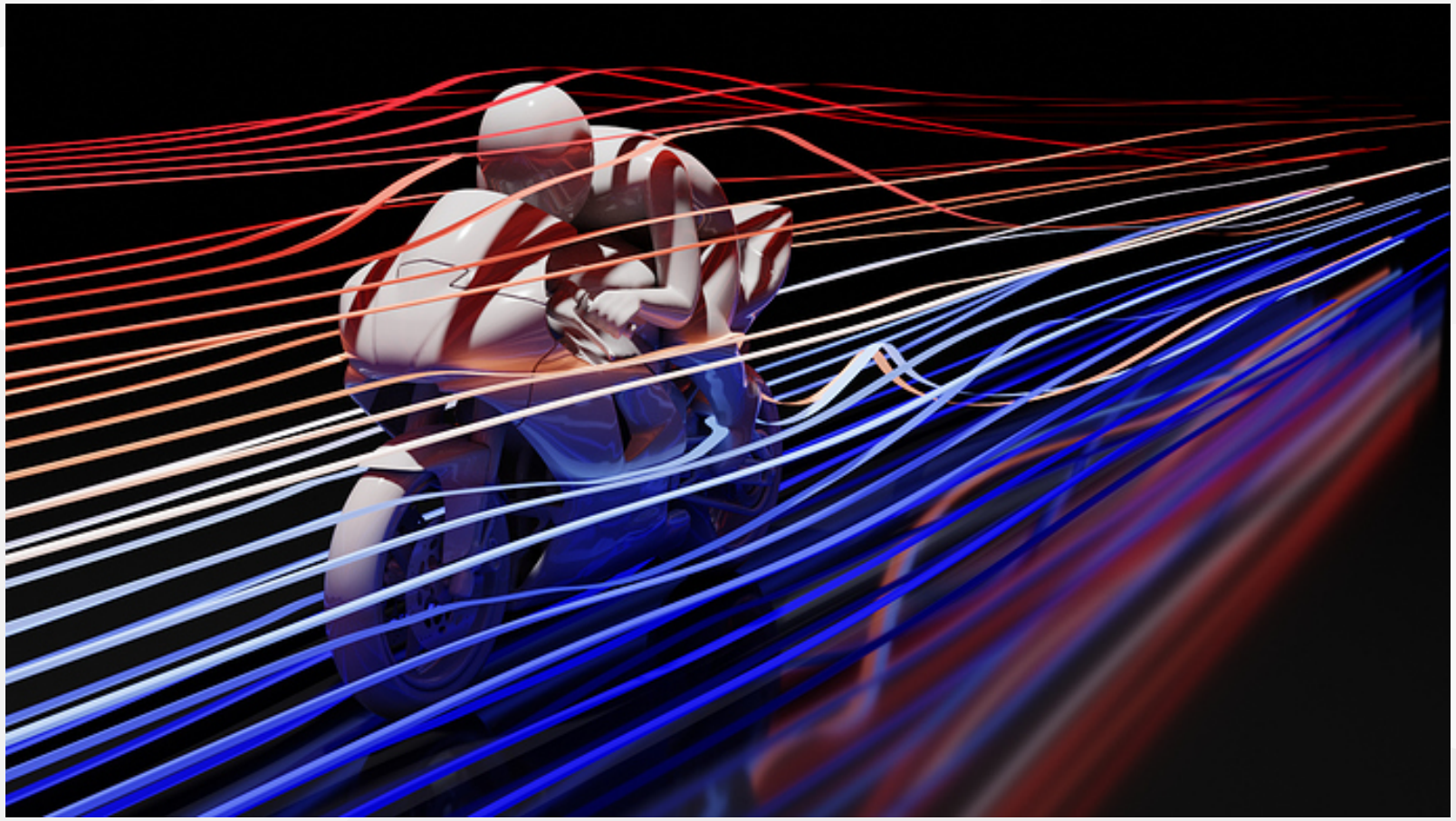
source: Silvano Imboden, 2020

Example



source: [Lemaire, Sebastien (2023)]

Example



source: [Tuomo Keskitalo using BVtkNodes]

Example

- FlipFluid demo reel

Example

- https://blenderartists.org/t/bvtknodes-gallery/1161079/77

General workflow

- Run simulation
- Postprocess it (Paraview for example)
- Export intermediary files
- Import them in Blender
- Design and setup the render
- Generate renders (single image or animation)

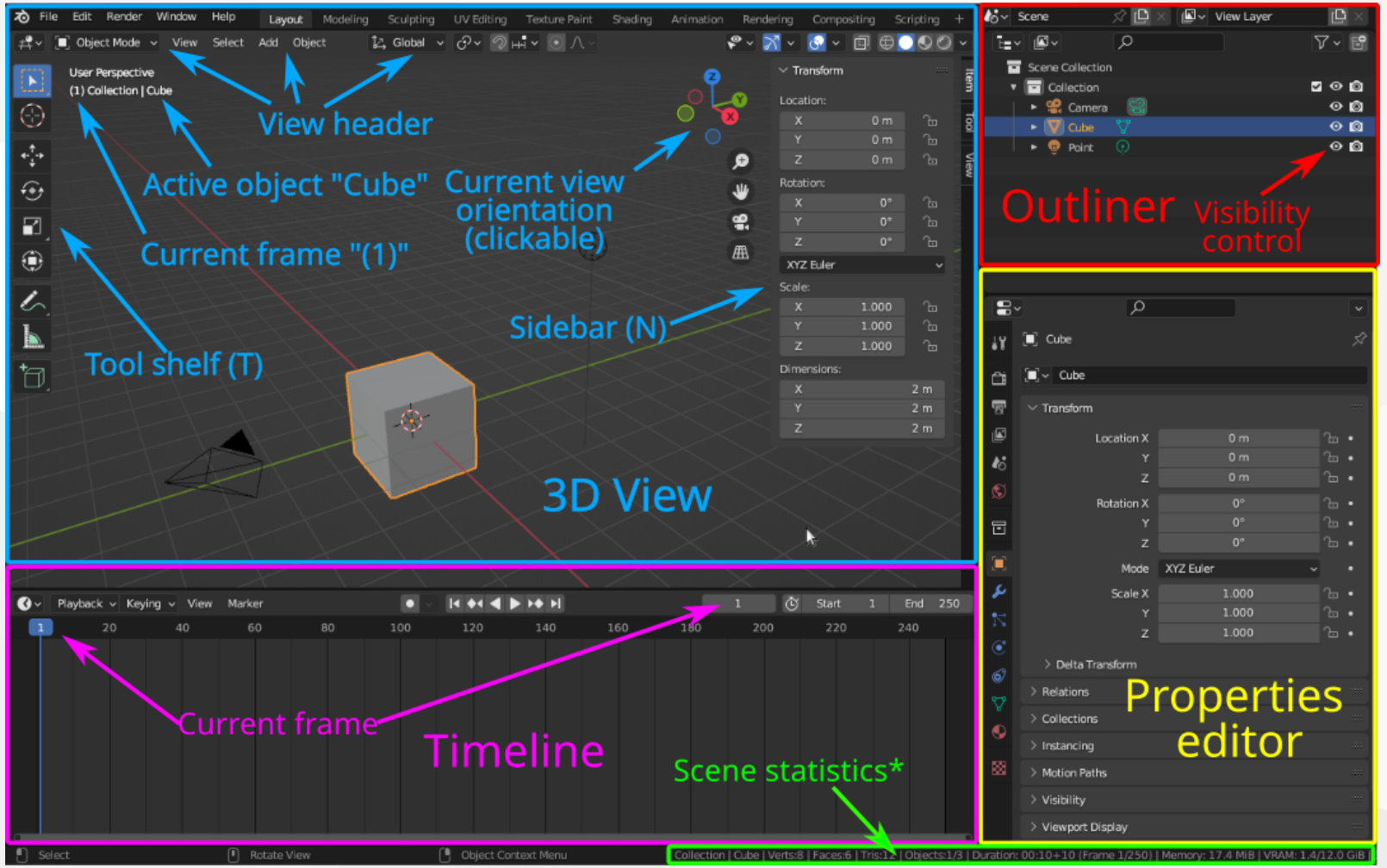
File Import

Blender can read a wide range of files, mainly:

`.ply`	`.x3d`	`.vdb`
single surface	multiple surfaces	volume
binary file	ASCII (XML) file	binary file
	slow import time	
single color/vertex	single color/vertex	multiple fields

- but also `.dae`, `.abc`, `.usd`, `.obj`, `.stl`, `.fbx`, `.gltf` ...
- external addons like: `bVTKnode`

Blender interface



source: SURF High-Performance Computing

Designing a scene

- Lights
  - Type: point, area, sun
  - Intensity
  - or HDRi 360 image
- Materials
  - Colour, roughness,
  - Complex textures with node setup
- Camera
  - Position, aspect ratio, resolution
  - Focal length, Aperture, Focus point

Rendering

- Setting up the Engine -> `Cycles`
- Output resolution and number of samples
- Rendering backend
  - CPU
  - GPU (CUDA, optiX, HIP or oneAPI)
- On your local machine -> `F12`

HPC specific workflow

- Get Blender from https://www.blender.org/download/
- Running headless

```
blender -b scene.blend -o "export-###.png" -E CYCLES -f 1 -- --cycles-device CPU
```

- Devices available: `CPU`, `CUDA`, `OPTIX`, `HIP`, `ONEAPI` and `METAL`
- https://docs.blender.org/manual/en/latest/advanced/command_line/arguments.html

Render animations

- Generate intermediary files (.x3d, .ply etc.)
- Rendering: use Blender's python API to load them for each timestep

```
import bpy

...
bpy.ops.import_scene.x3d(filepath=x3d_filename)
objs['Shape_IndexedFaceSet'].material_slots['Shape'].material = bpy.data.materials['Qcrit']
bpy.ops.render.render(write_still=True)
```

```
blender -b scene_anim.blend -P loader.py ...
```

- https://docs.blender.org/api/current

Performance considerations

- Single node only
  - CPU: scales well with core count
  - GPU:
    - can run on multiple GPUs
    - doesn't scale idealy
    - with animations, better to distribute load on multiple single GPU instances
- Benchmarks data: https://opendata.blender.org/benchmarks/query

Resources

- **Introduction to Scientific Visualisation with Blender**: free MOOC by Surf
  - Detailed course with example files, videos, cheat sheets etc.
- Cinematic Scientific Visualization: Where Science Meets Hollywood Visual Effects: presentation by Kalina Borkiewicz
- 3D Data Visualisation for Science Communication: free MOOC by Advanced Visualization Lab at NCSA
- BlenderGuru: Video tutorials on youtube

Questions?

Sébastien Lemaire (s.lemaire@epcc.ed.ac.uk)