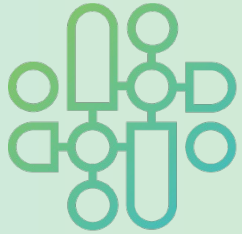


SC23
Denver, CO | i am hpc.

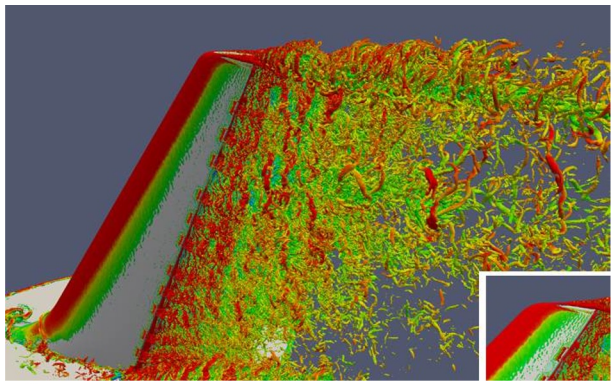
In Situ Analysis and Visualization with Ascent and ParaView Catalyst

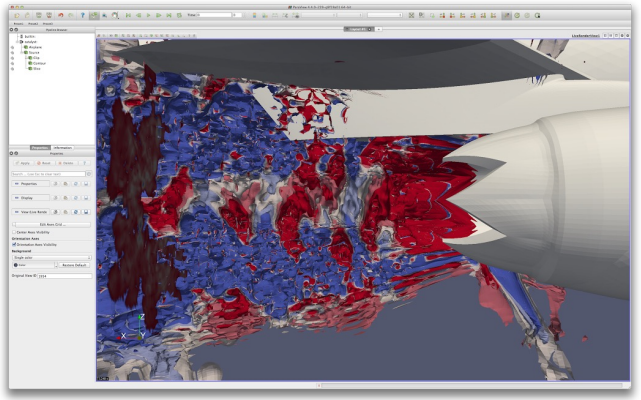
[Catalyst Hands-on]

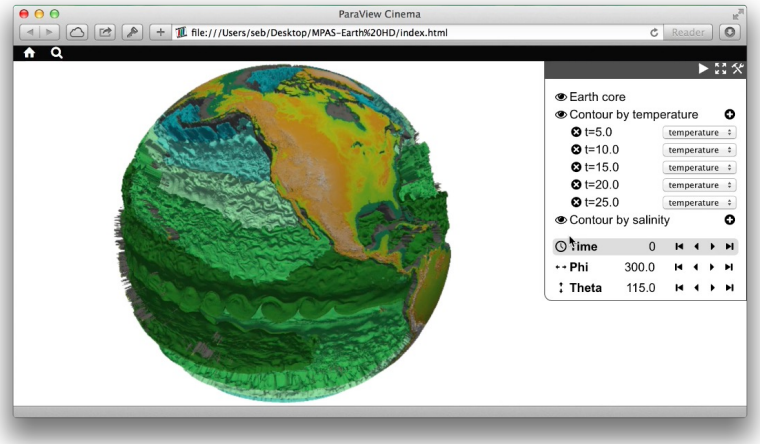
SC23 Tutorial
Monday November 13th, 2023

Corey Wetterer-Nelson, Kitware Inc.

Catalyst







kitware

Anatomy of a Catalyst Adaptor

https://gitlab.kitware.com/paraview/paraview/-/tree/master/Examples/Catalyst2

- Image Data
- Polygonal
- Polyhedra
- Multimesh
- Multi Channel

Anatomy of a Catalyst Adaptor

Initialize

CxxFullExample — vi CatalystAdaptor.h — 84×16

```
void Initialize(int argc, char* argv[])
{
  conduit_cpp::Node node;
  for (int cc = 1; cc < argc; ++cc)
  {
    node["catalyst/scripts/script" + std::to_string(cc - 1)].set_string(argv[cc]);
  }
  node["catalyst_load/implementation"] = "paraview";
  node["catalyst_load/search_paths/paraview"] = PARAVIEW_IMPL_DIR;
  catalyst_status err = catalyst_initialize(conduit_cpp::c_node(&node));
  if (err != catalyst_status_ok)
  {
    std::cerr << "Failed to initialize Catalyst: " << err << std::endl;
  }
}
```

kitware

Anatomy of a Catalyst Adaptor

Execute

CxxFullExample — vi CatalystAdaptor.h — 70×12

```
void Execute(int cycle, double time, Grid& grid, Attributes& attribs)
{
  conduit_cpp::Node exec_params;

  // add time/cycle information
  auto state = exec_params["catalyst/state"];
  state["timestep"].set(cycle);
  state["time"].set(time);
  state["multiblock"].set(1);
```

kitware

Anatomy of a Catalyst Adaptor

Execute

CxxFullExample — vi CatalystAdaptor.h — 99×29

```
// Add channels.
// We only have 1 channel here. Let's name it 'grid'.
auto channel = exec_params["catalyst/channels/grid"];

// Since this example is using Conduit Mesh Blueprint to define the mesh,
// we set the channel's type to "mesh".
channel["type"].set("mesh");

// now create the mesh.
auto mesh = channel["data"];

// start with coordsets (of course, the sequence is not important, just make
// it easier to think in this order).
mesh["coordsets/coords/type"].set("explicit");

mesh["coordsets/coords/values/x"].set_external(
  grid.GetPointsArray(), grid.GetNumberOfPoints(), /*offset=*/0, /*stride=*/3 * sizeof(double));
mesh["coordsets/coords/values/y"].set_external(grid.GetPointsArray(), grid.GetNumberOfPoints(),
  /*offset=*/sizeof(double), /*stride=*/3 * sizeof(double));
mesh["coordsets/coords/values/z"].set_external(grid.GetPointsArray(), grid.GetNumberOfPoints(),
  /*offset=*/2 * sizeof(double), /*stride=*/3 * sizeof(double));

// Next, add topology
mesh["topologies/mesh/type"].set("unstructured");
mesh["topologies/mesh/coordset"].set("coords");
mesh["topologies/mesh/elements/shape"].set("hex");
mesh["topologies/mesh/elements/connectivity"].set_external(
  grid.GetCellPoints(0), grid.GetNumberOfCells() * 8);
```

kitware

Anatomy of a Catalyst Adaptor

Execute

CxxFullExample — vi CatalystAdaptor.h — 111×27

```
  // Finally, add fields.
  auto fields = mesh["fields"];
  fields["velocity/association"].set("vertex");
  fields["velocity/topology"].set("mesh");
  fields["velocity/volume_dependent"].set("false");

  // velocity is stored in non-interlaced form (unlike points).
  fields["velocity/values/x"].set_external(
    attribs.GetVelocityArray(), grid.GetNumberOfPoints(), /*offset=*/0);
  fields["velocity/values/y"].set_external(attribs.GetVelocityArray(), grid.GetNumberOfPoints(),
    /*offset=*/grid.GetNumberOfPoints() * sizeof(double));
  fields["velocity/values/z"].set_external(attribs.GetVelocityArray(), grid.GetNumberOfPoints(),
    /*offset=*/grid.GetNumberOfPoints() * sizeof(double) * 2);

  // pressure is cell-data.
  fields["pressure/association"].set("element");
  fields["pressure/topology"].set("mesh");
  fields["pressure/volume_dependent"].set("false");
  fields["pressure/values"].set_external(attribs.GetPressureArray(), grid.GetNumberOfCells());

  catalyst_status err = catalyst_execute(conduit_cpp::c_node(&exec_params));
  if (err != catalyst_status_ok)
  {
    std::cerr << "Failed to execute Catalyst: " << err << std::endl;
  }
}
```

kitware

Anatomy of a Catalyst Adaptor

Finalize

CxxFullExample — vi CatalystAdaptor.h — 71×11

```
void Finalize()
{
  conduit_cpp::Node node;
  catalyst_status err = catalyst_finalize(conduit_cpp::c_node(&node));
  if (err != catalyst_status_ok)
  {
    std::cerr << "Failed to finalize Catalyst: " << err << std::endl;
  }
}
```

kitware

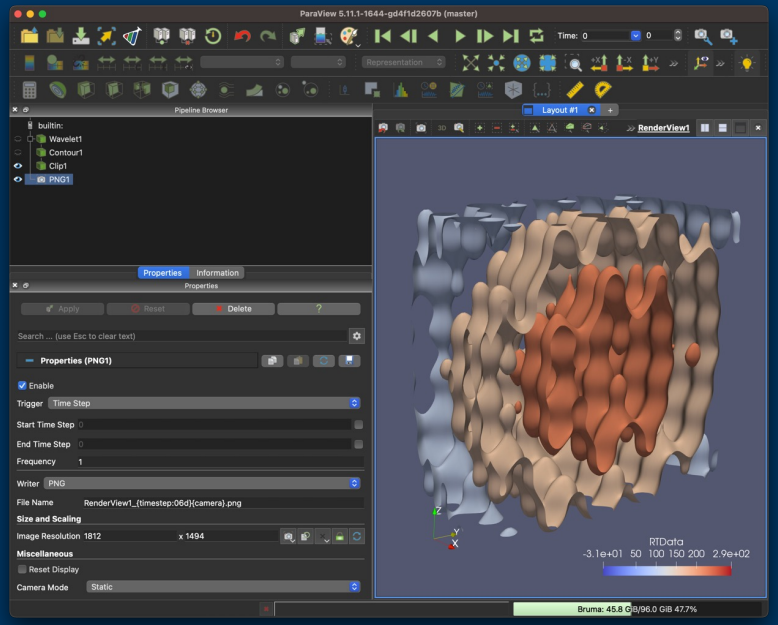
Customizing Catalyst Pipelines

- Build example with Catalyst enabled
- Run example with sample pipeline script
- Modify the script to manipulate the pipeline
- Rerun the example with your new script

ParaView Show & Tell

Follow along!

https://www.paraview.org/download/



kitware