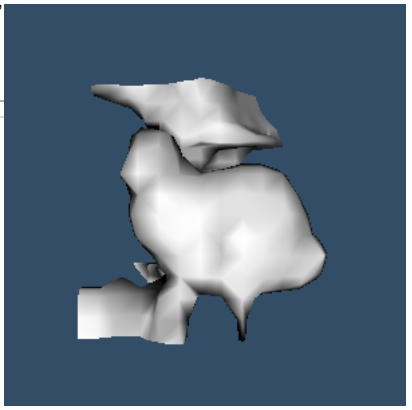
VTK/Examples/Cxx/Filtering/SurfaceFromUnorganizedPoints - KitwarePublic

VTK/Examples/Cxx/Filtering/SurfaceFromUnorganizedPoints

< VTK | Examples | Cxx

This example creates points on a sphere and then finds the surface through the points. If an optional polydata file is provided, the example operates on the points in that polydata.



SurfaceFromUnorganizedPoints.cxx

```
#include <vtkVersion.h>
#include <vtkSmartPointer.h>

#include <vtkSurfaceReconstructionFilter.h>
#include <vtkProgrammableSource.h>
#include <vtkContourFilter.h>
#include <vtkReverseSense.h>
#include <vtkPolyDataMapper.h>
#include <vtkProperty.h>
#include <vtkPolyData.h>
#include <vtkCamera.h>
#include <vtkRenderer.h>
#include <vtkRenderWindow.h>
#include <vtkRenderWindowInteractor.h>
#include <vtkSphereSource.h>
#include <vtkXMLPolyDataReader.h>

int main(int argc, char *argv[])
{

  vtkSmartPointer<vtkPolyData> input;
  if(argc > 1)
    {
    vtkSmartPointer<vtkXMLPolyDataReader> reader =
      vtkSmartPointer<vtkXMLPolyDataReader>::New();
    reader->SetFileName(argv[1]);
    reader->Update();
    input = reader->GetOutput();
    }
  else
    {
    vtkSmartPointer<vtkSphereSource> sphereSource =
      vtkSmartPointer<vtkSphereSource>::New();
    sphereSource->Update();
    input = sphereSource->GetOutput();
    }

  vtkSmartPointer<vtkPolyData> polydata =
    vtkSmartPointer<vtkPolyData>::New();
  polydata->SetPoints(input->GetPoints());

  // Construct the surface and create isosurface.
  vtkSmartPointer<vtkSurfaceReconstructionFilter> surf =
    vtkSmartPointer<vtkSurfaceReconstructionFilter>::New();
#if VTK_MAJOR_VERSION <= 5
  surf->SetInput(polydata);
#else
  surf->SetInputData(polydata);
#endif

  vtkSmartPointer<vtkContourFilter> cf =
    vtkSmartPointer<vtkContourFilter>::New();
  cf->SetInputConnection(surf->GetOutputPort());
  cf->SetValue(0, 0.0);

  // Sometimes the contouring algorithm can create a volume whose gradient
  // vector and ordering of polygon (using the right hand rule) are
  // inconsistent. vtkReverseSense cures this problem.
  vtkSmartPointer<vtkReverseSense> reverse =
    vtkSmartPointer<vtkReverseSense>::New();
  reverse->SetInputConnection(cf->GetOutputPort());
  reverse->ReverseCellsOn();
  reverse->ReverseNormalsOn();

  vtkSmartPointer<vtkPolyDataMapper> map =
    vtkSmartPointer<vtkPolyDataMapper>::New();
  map->SetInputConnection(reverse->GetOutputPort());
  map->ScalarVisibilityOff();

  vtkSmartPointer<vtkActor> surfaceActor =
    vtkSmartPointer<vtkActor>::New();
  surfaceActor->SetMapper(map);

  // Create the RenderWindow, Renderer and both Actors
  vtkSmartPointer<vtkRenderer> ren =
    vtkSmartPointer<vtkRenderer>::New();
  vtkSmartPointer<vtkRenderWindow> renWin =
    vtkSmartPointer<vtkRenderWindow>::New();
  renWin->AddRenderer(ren);
  vtkSmartPointer<vtkRenderWindowInteractor> iren =
    vtkSmartPointer<vtkRenderWindowInteractor>::New();
  iren->SetRenderWindow(renWin);
```

```
  // Add the actors to the renderer, set the background and size
  ren->AddActor(surfaceActor);
  ren->SetBackground(.2, .3, .4);

  renWin->Render();
  iren->Start();

  return EXIT_SUCCESS;

}
```

CMakeLists.txt

```
cmake_minimum_required(VERSION 2.8)

PROJECT(SurfaceFromUnorganizedPoints)

find_package(VTK REQUIRED)
include(${VTK_USE_FILE})

add_executable(SurfaceFromUnorganizedPoints MACOSX_BUNDLE SurfaceFromUnorganizedPoints)

if(VTK_LIBRARIES)
  target_link_libraries(SurfaceFromUnorganizedPoints ${VTK_LIBRARIES})
else()
  target_link_libraries(SurfaceFromUnorganizedPoints vtkHybrid vtkWidgets)
endif()
```

Download and Build SurfaceFromUnorganizedPoints

Click here to download SurfaceFromUnorganizedPoints. and its CMakeLists.txt file.

Once the tarball SurfaceFromUnorganizedPoints.tar has been downloaded and extracted,

```
cd SurfaceFromUnorganizedPoints/build
```

- If VTK is installed:

```
cmake ..
```

- If VTK is not installed but compiled on your system, you will need to specify the path to your VTK build:

```
cmake -DVTK_DIR:PATH=/home/me/vtk_build ..
```

Build the project:

```
make
```

and run it:

```
./SurfaceFromUnorganizedPoints
```

WINDOWS USERS PLEASE NOTE: Be sure to add the VTK bin directory to your path. This will resolve the VTK dll's at run time.