

ElVis – The Element Visualizer

Usual visualization tools (Tecplot, Paraview...) use linear piecewise interpolation to display simulation results.

To display high-order finite element solutions, these tools introduce errors into the final image.

ElVis is an integrated visualization system designed specifically for high-order finite element solutions.

ElVis – The Element Visualizer

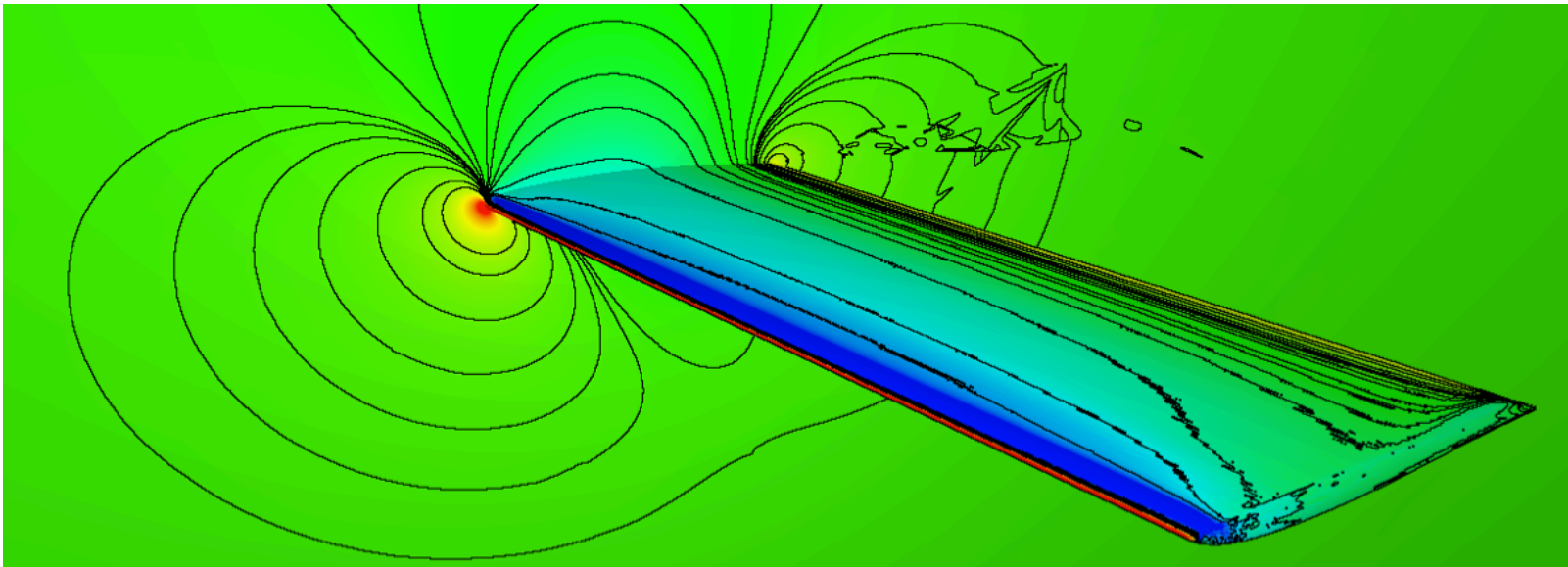
Extensible Architecture:
allows new high-order
systems to be added to
ElVis with minimal effort.

Accurate Visualization:
avoids errors into the
final image by operating
on the high-order data
directly.

Interactive Performance:
uses the parallel
processing capabilities of
recent Graphics
Processing Units (GPU).

ElVis – The Element Visualizer

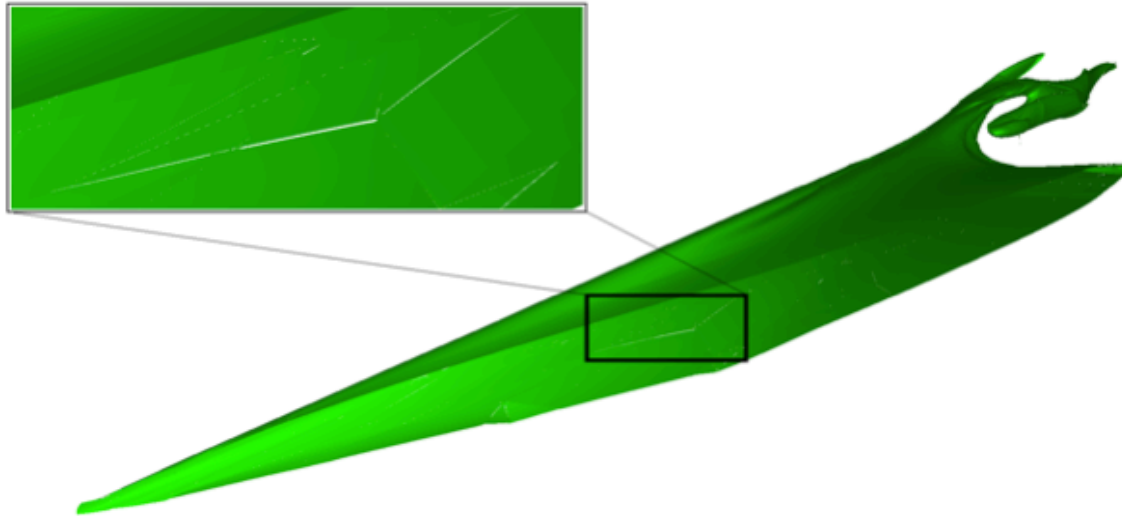
Cut surface visualization and contouring: ElVis uses the high-order data in its native state (without the need for low-order approximation) and the mathematical properties of the underlying fields to provide accurate images.



Density field on the ONERA M6 wing. Contours on curved and planar surfaces.

ElVis – The Element Visualizer

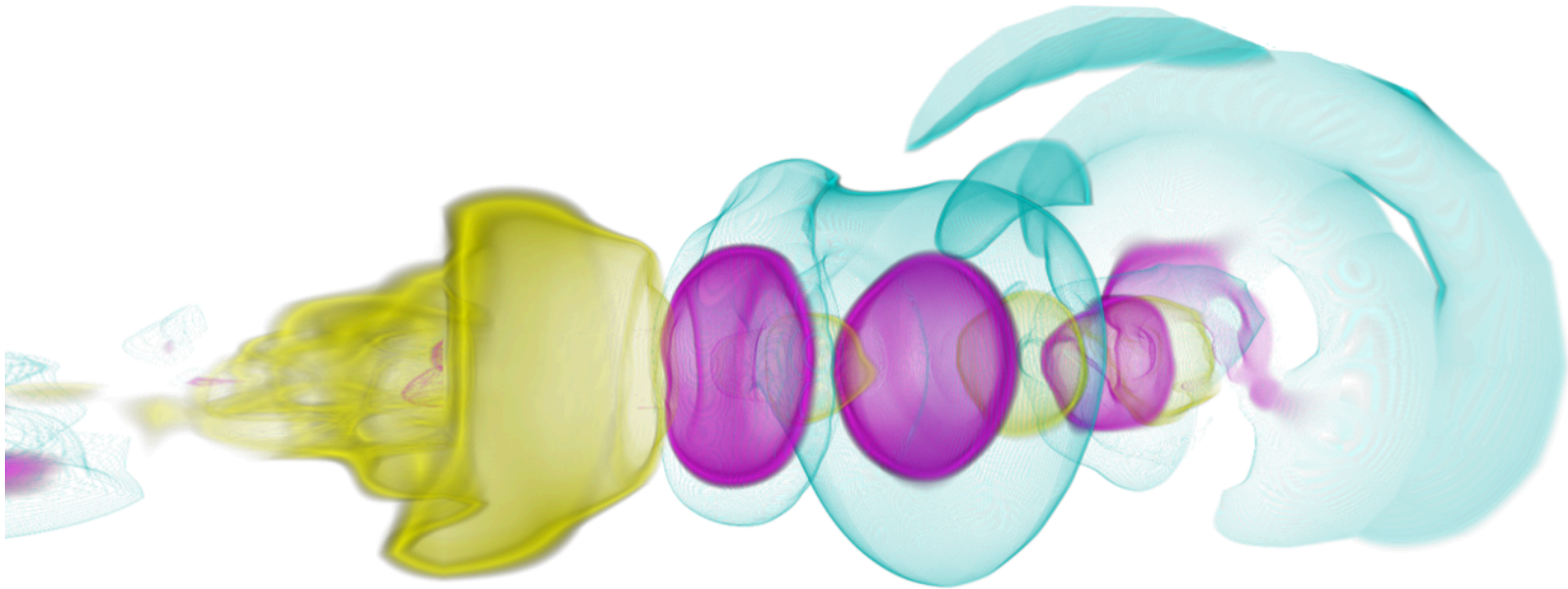
Isosurfaces: high-order isosurface rendering algorithms respect the features of the high-order data (which is not the case of existing “object-space” methods). Avoid to miss valid features of DG simulations.



Example of crack in the surface that arise for “object-space” method when the underlying solution is from a DG method.

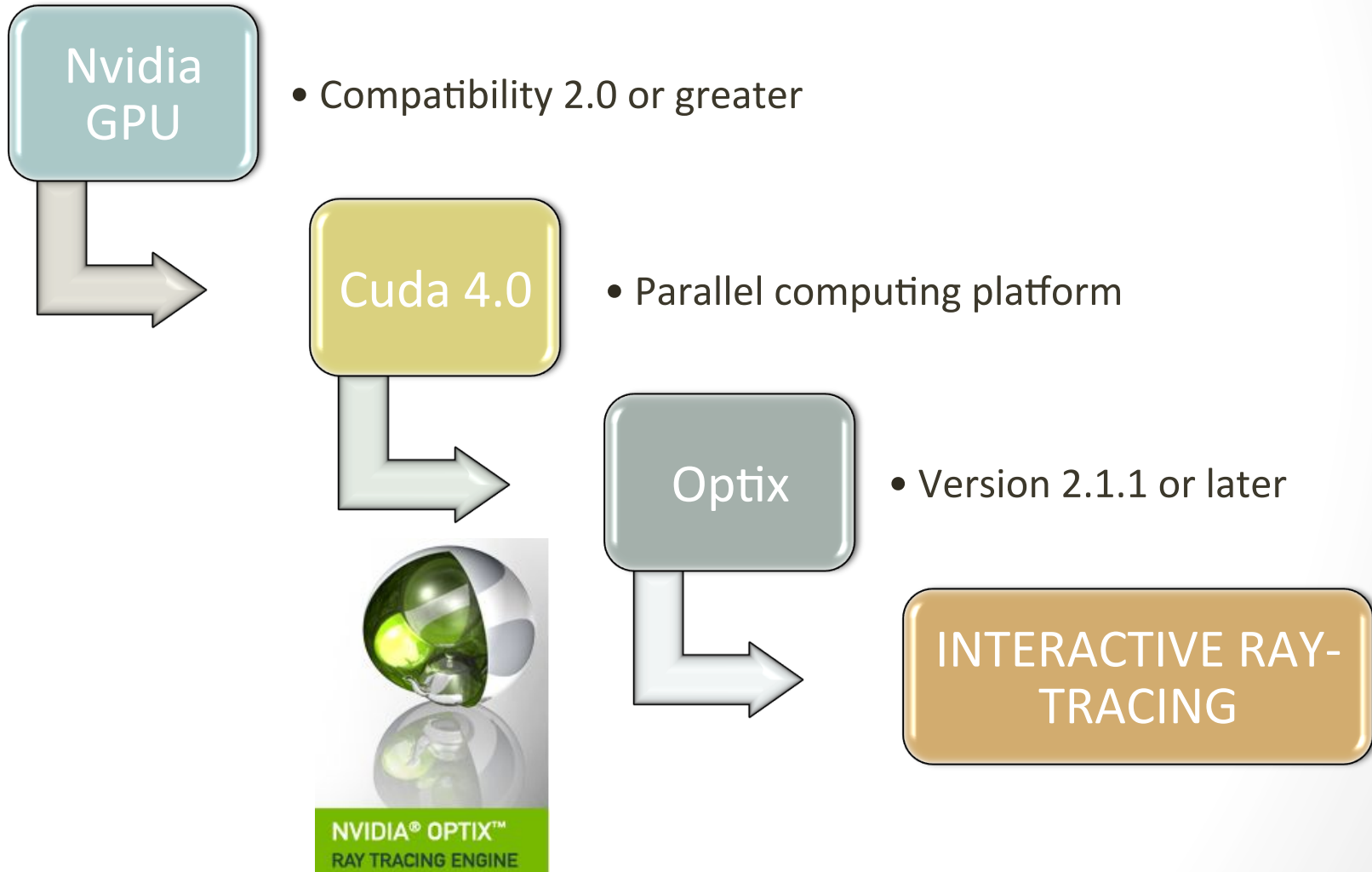
ElVis – The Element Visualizer

Volume Rendering: ElVis implements a direct volume rendering method that uses the properties of spectral/hp finite element fields to produce images that are both accurate and interactive.



Volume rendering of the pressure of a high-order finite element solution of fluid flow past a rotating canister.

System Requirements



Ray-tracing

Generating an image by tracing the path of light through pixels in an image plane.

ELVis implements interactive ray-tracing algorithm to evaluate the solution at each pixel.



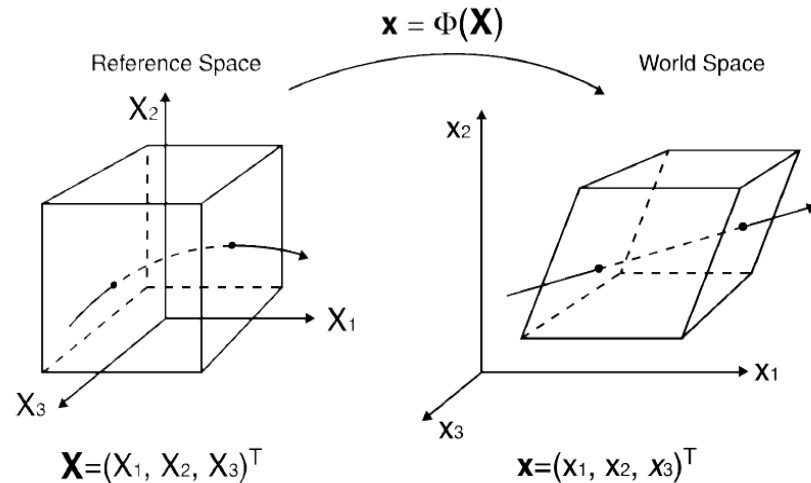
Ray tracing can create realistic images

Pixel-Exact Images

1 - When a ray hits a pixel, the element hit by this ray is identified and the coordinates of the intersection point (in the World space) are stored.

2 - A Newton's method implements the inverse mapping function to find the equivalent of the intersection point into the Reference element.

3 - These coordinates into the Reference element, the coefficients (provided by Nektar++) and the expansion basis are used to obtain a high-order approximation of the field at the intersection point into the World space.



4 - This result is used to determine the colour of the corresponding pixel (effects of lights and shadows of the scene are also taken into account).

Nektar++ Extension

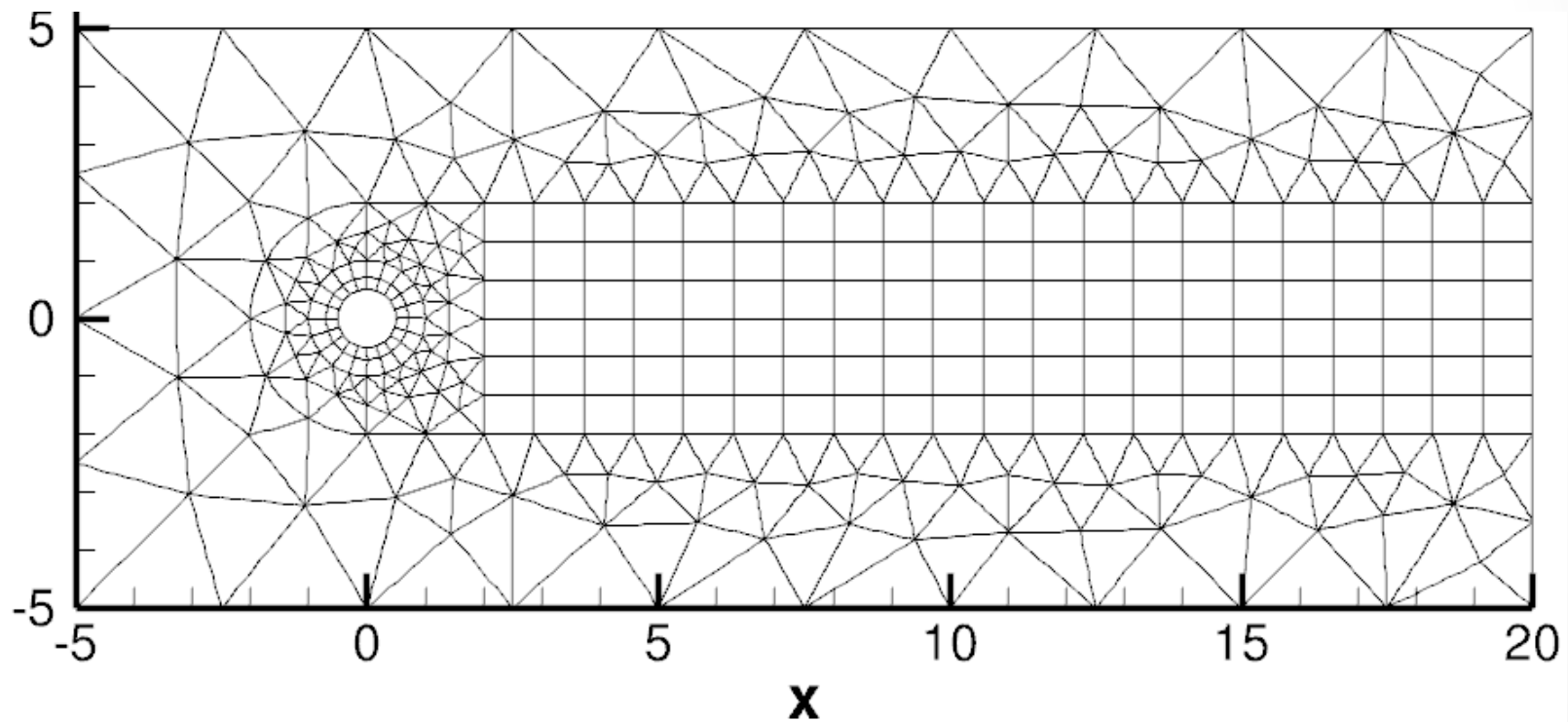
High-order visualization of native Nektar++ output files.

Joaquim Peiro's objective: using ELVis as an integrated pre- and post-processing tool for Nektar++.

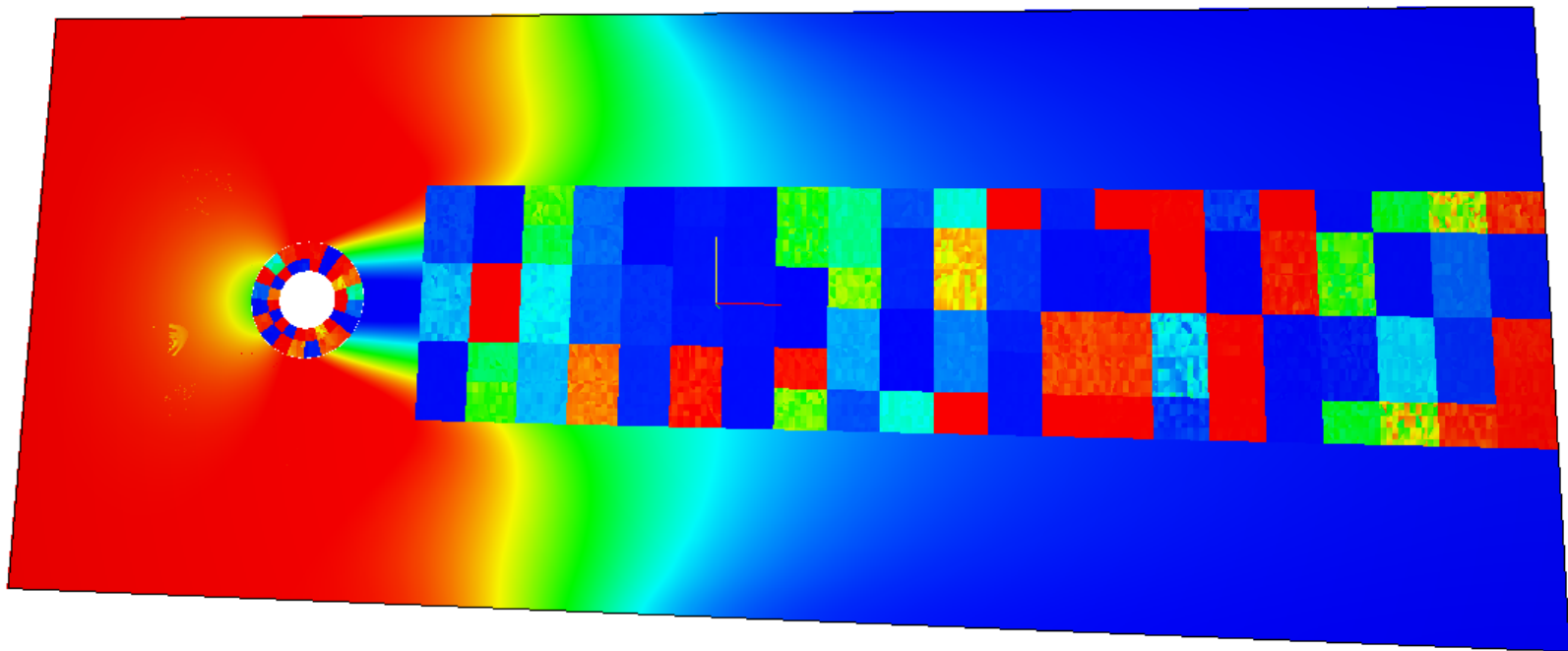
Elements currently supported:

- Straight-sided hexahedrons
 - Curved quads
 - Curved triangles
- However some bugs may appear...

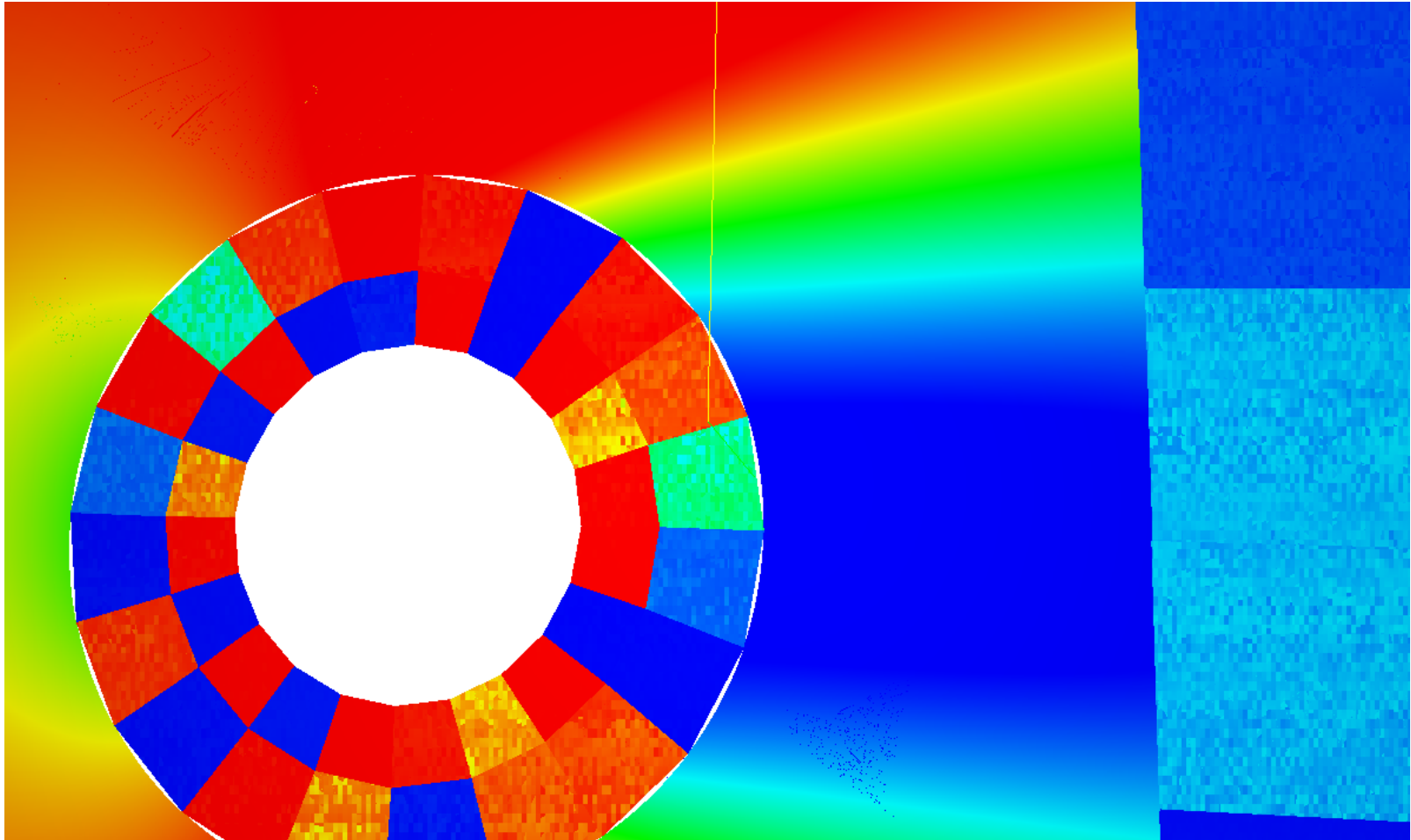
Nektar++ Extension



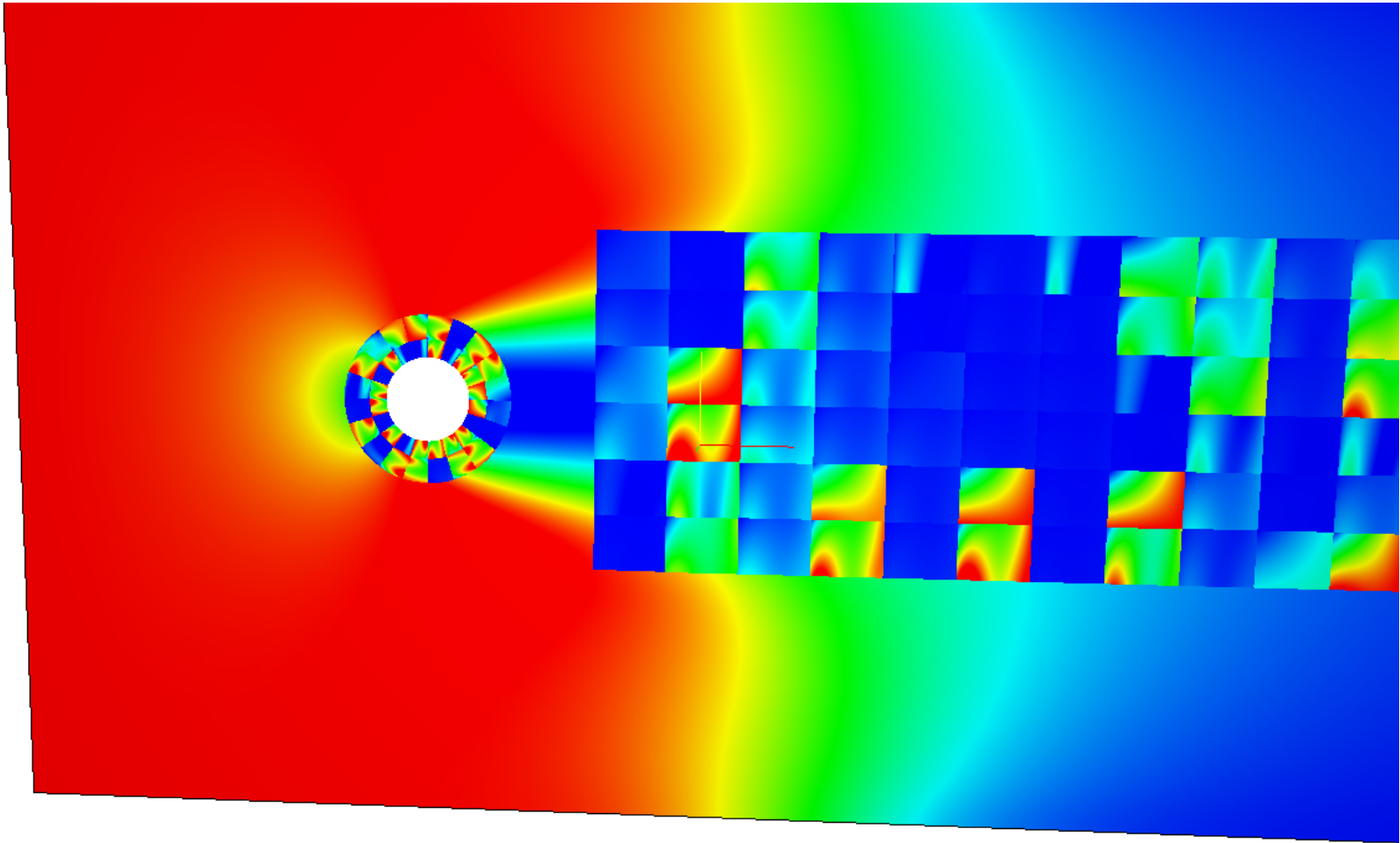
Nektar++ Extension



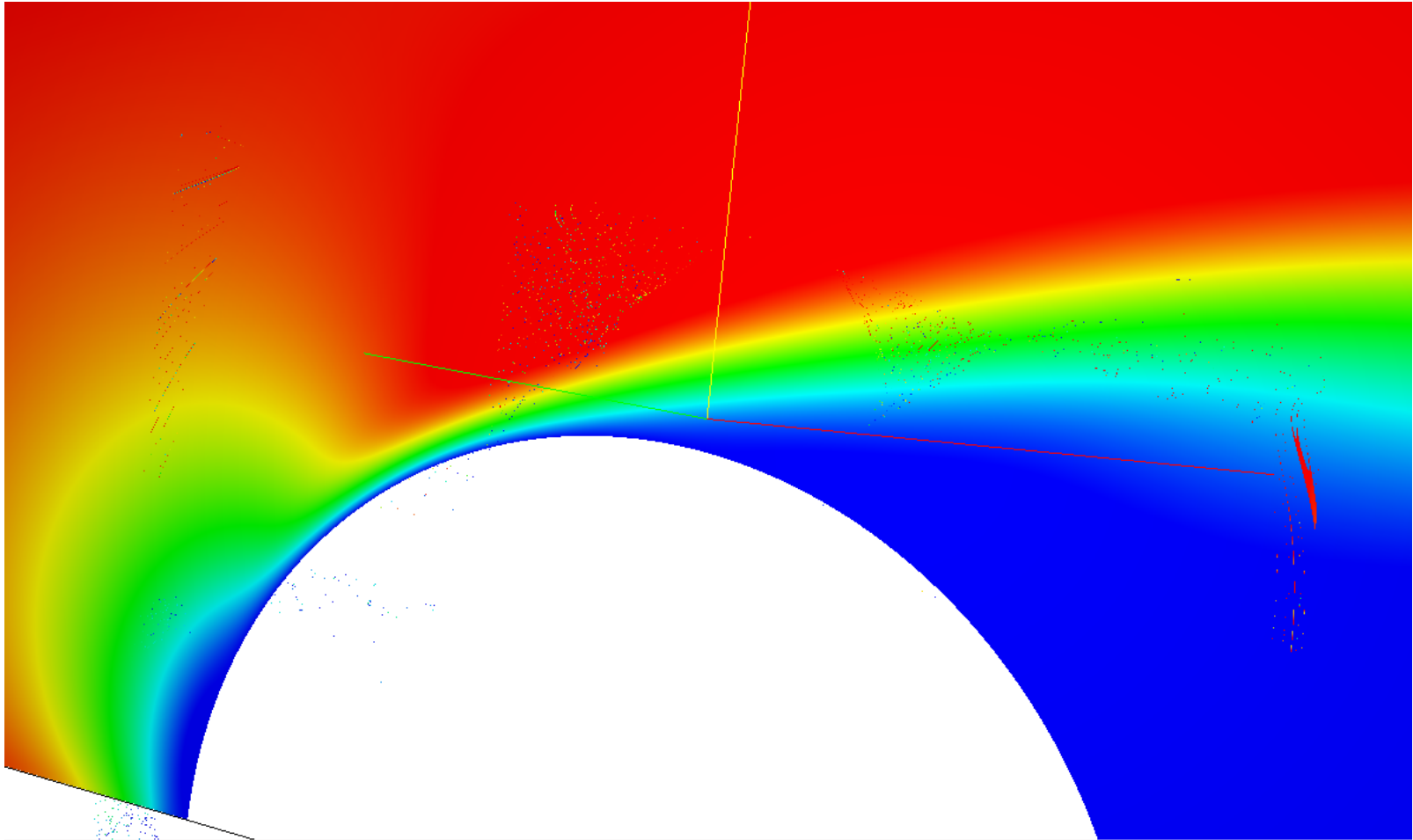
Nektar++ Extension



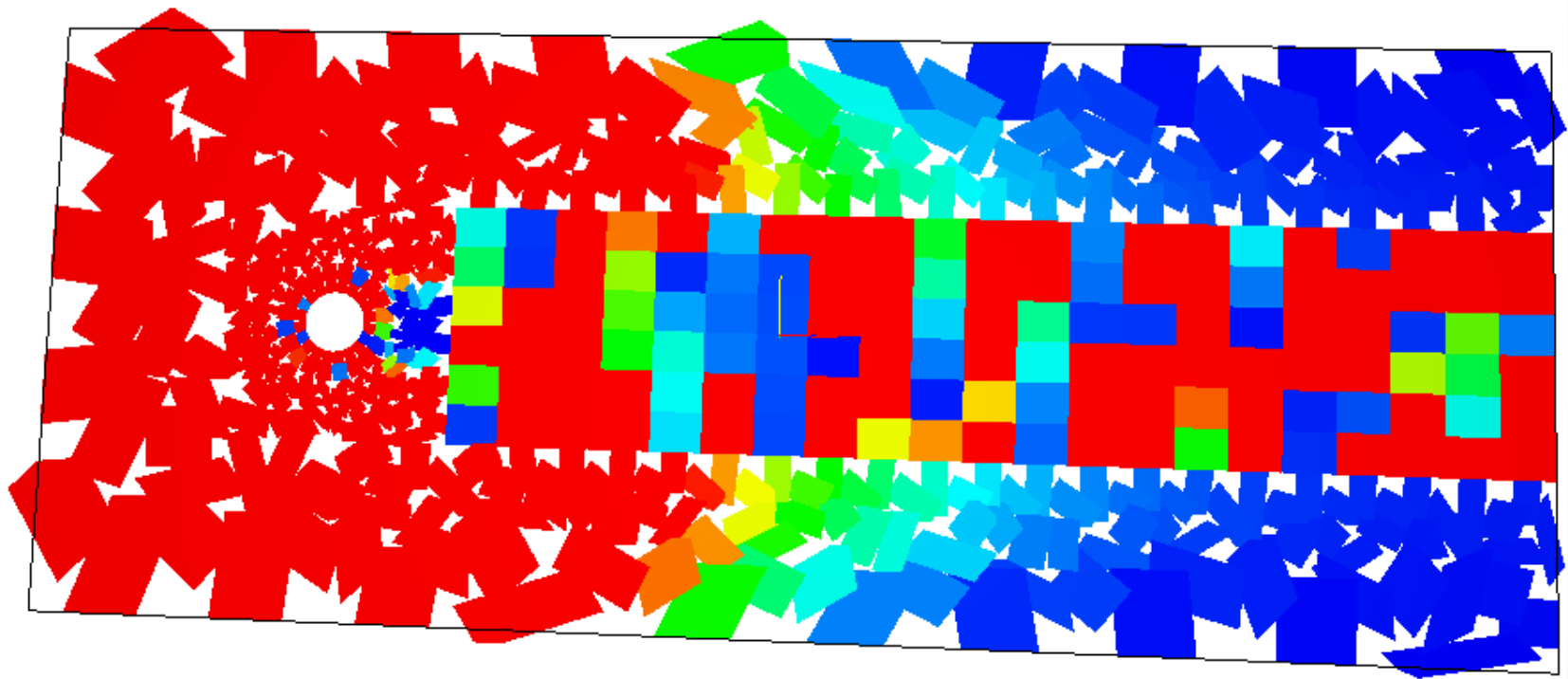
Nektar++ Extension



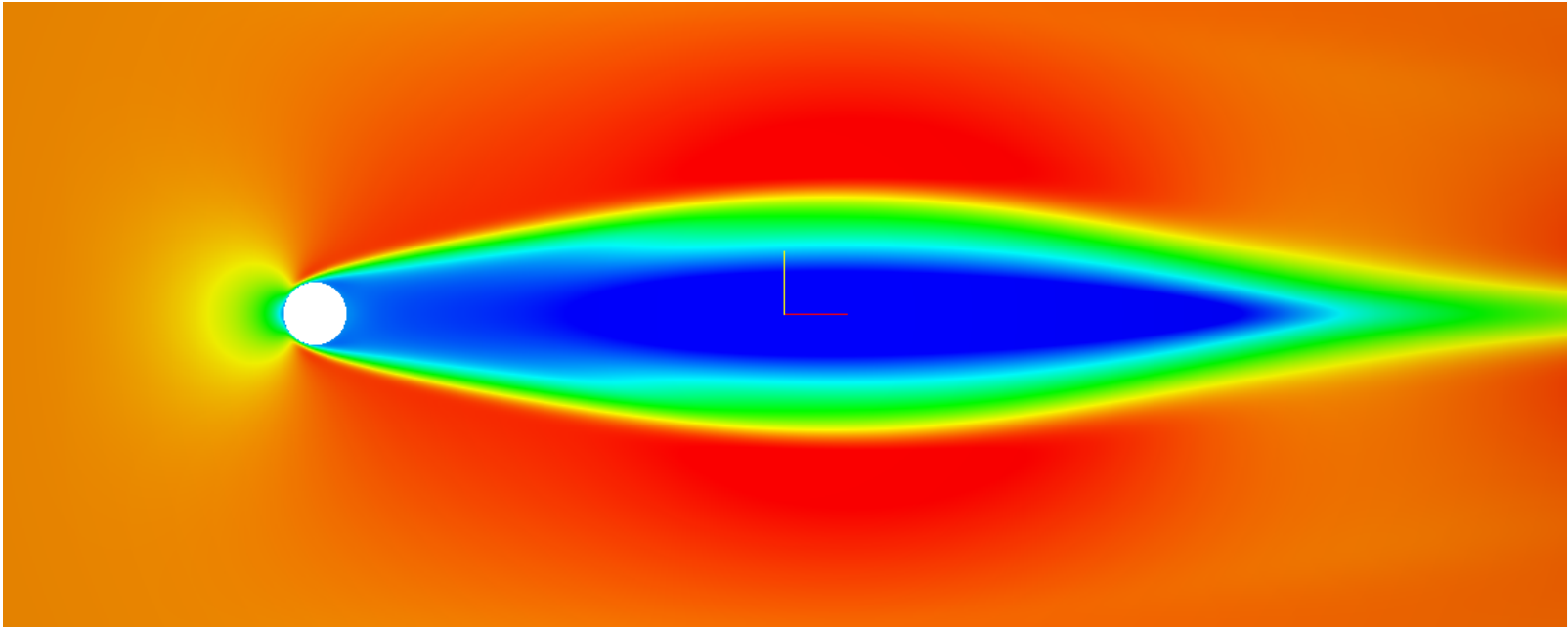
Nektar++ Extension



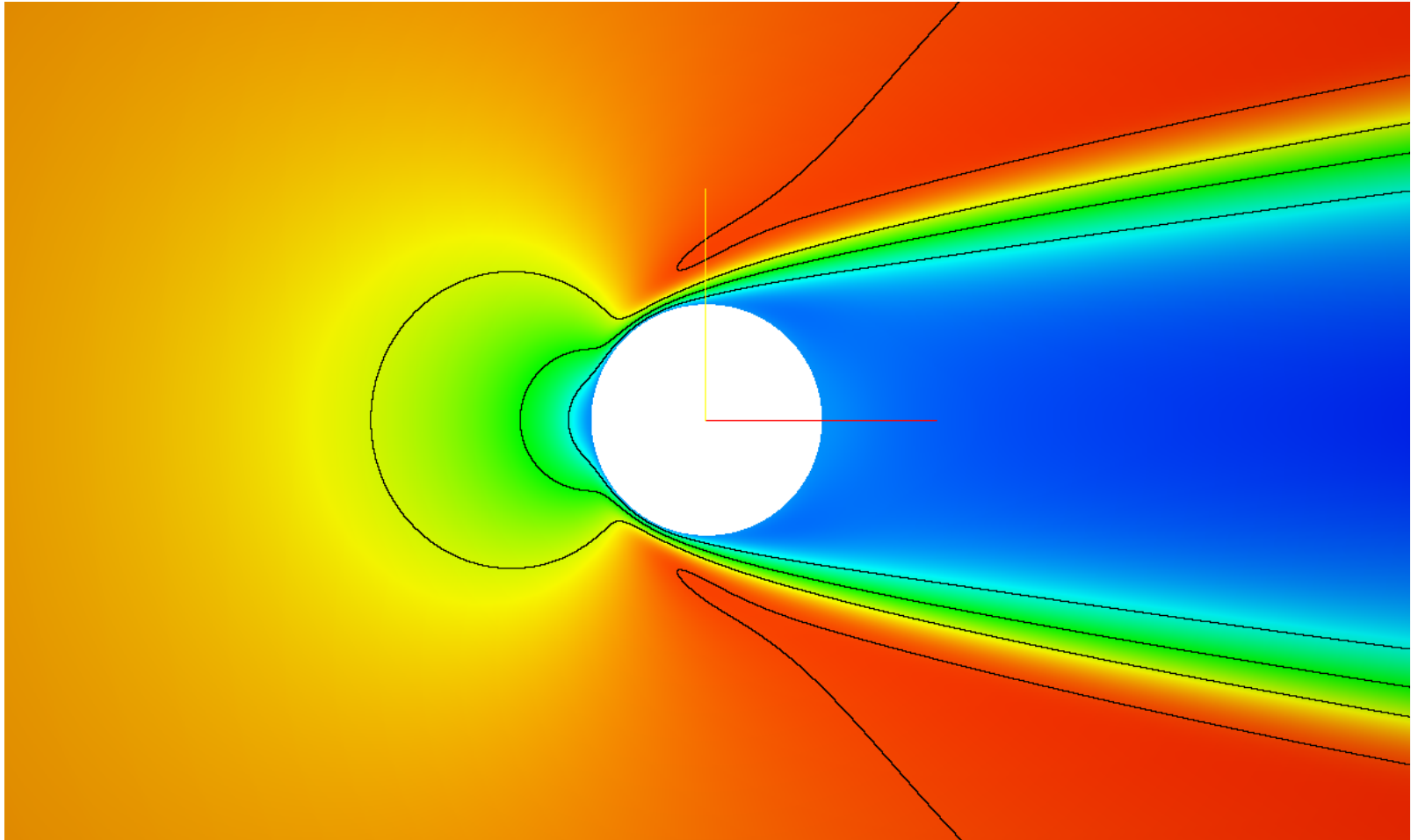
Nektar++ Extension



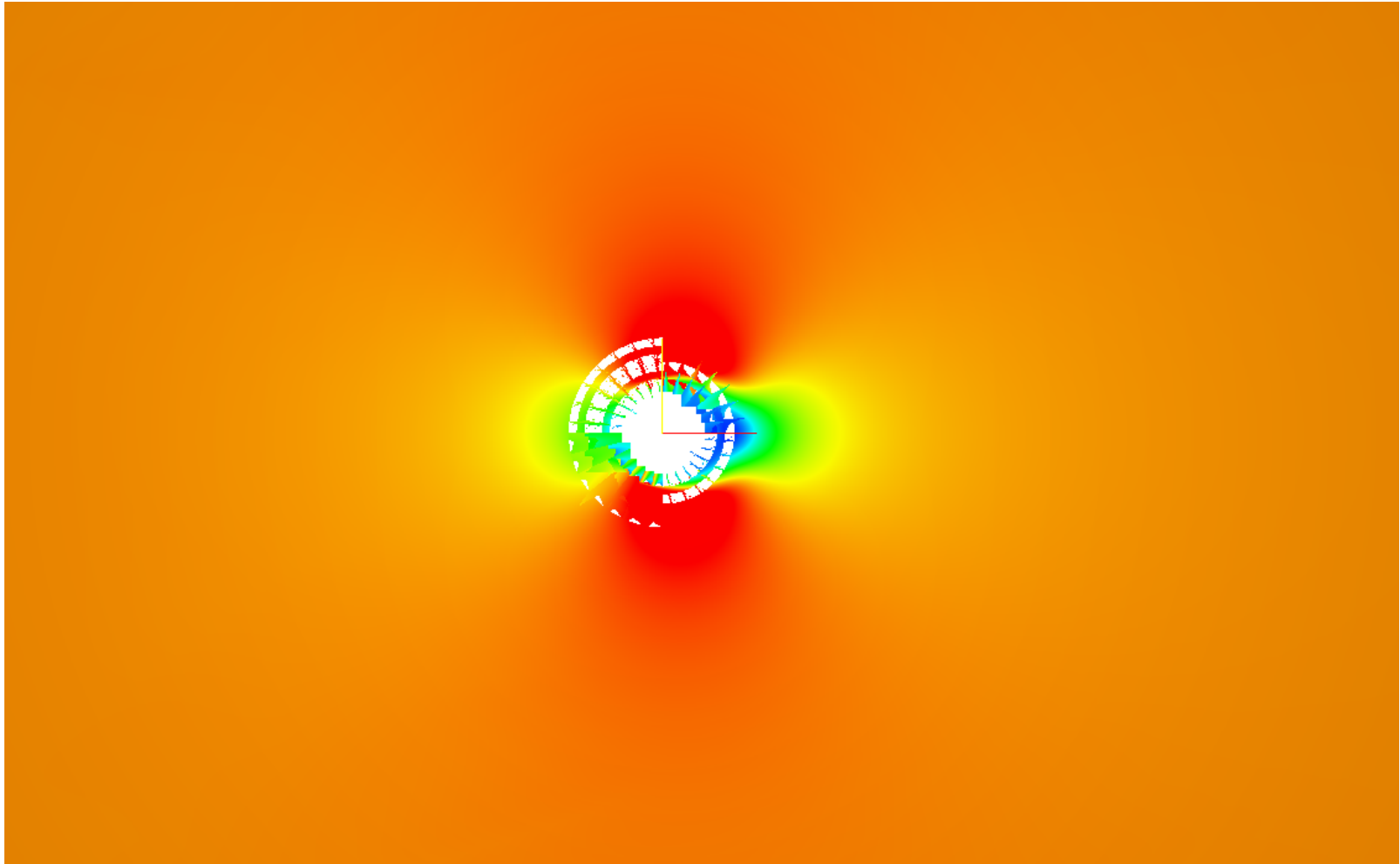
Nektar++ Extension



Nektar++ Extension



Nektar++ Extension



Nektar++ Extension

Use a "clever" initial guess for the Newton's method which implements the inverse mapping. This may reduce the number of the steps required to converge.

Implement curved hexahedrons.

Implement prisms and tetrahedrons.

ElVis enables to display results obtained only with the MODIFIED basis (GLL_LAGRANGE_SEM not supported for the moment).