

Real-time Web-Based Hemodynamic Visualization and Computation for 4D Flow MRI

Martin Cartajena

Vicomtech Foundation, BRTA
San Sebastián, Spain

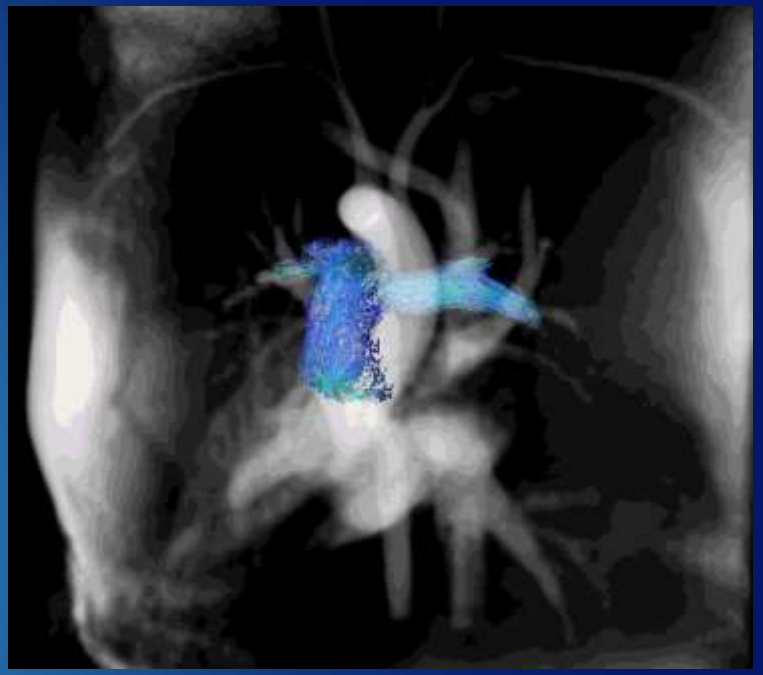
J. Bóbeda, A. Perez, W. Lainez-Oyuela, J. Ruíz-Cabello,
N. Arrarte, K. López-Linares and A. Arbelaiz

Web3D 2026
13–15 October 2026 · Doha, Qatar

4D Flow MRI of the pulmonary arteries

Magnitude series → **Anatomical context**

3 phase-contrast series x, y and z → **3D blood velocity over the cardiac cycle**

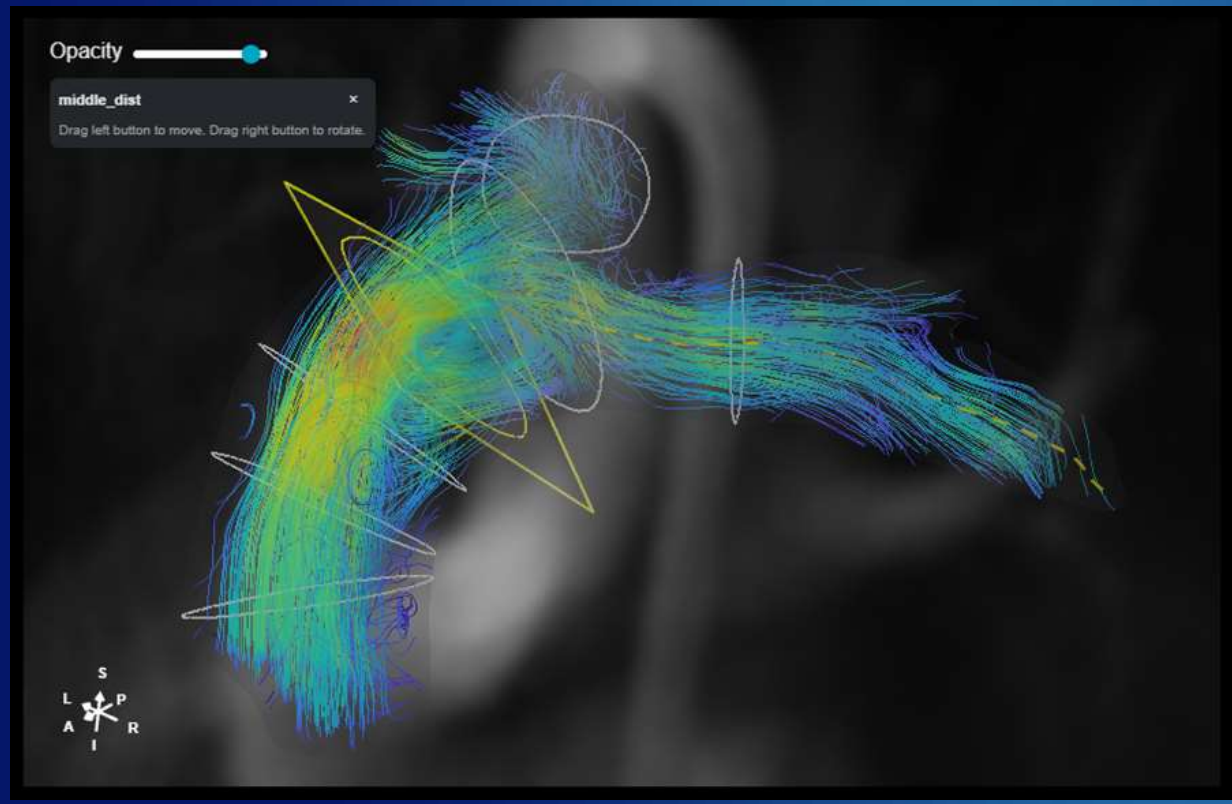


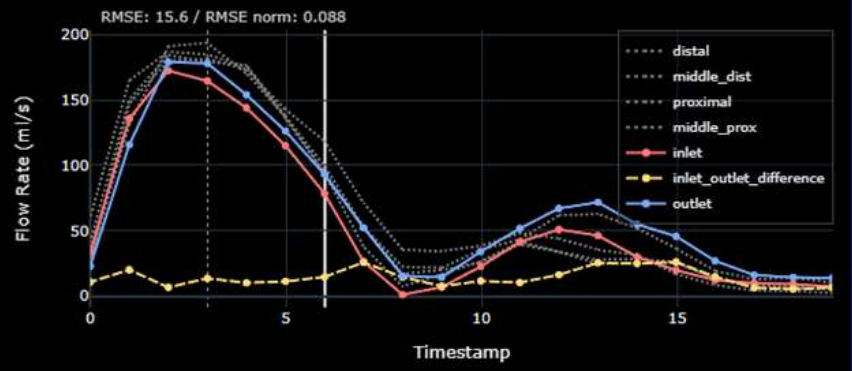
Pulmonary artery flow

20 cardiac frames

Anatomy and flow share the same physical coordinates

Interactive measurements at anatomical locations





Position	Area	mVma	mVm	pVma:	pVme:	FRmax	FRme:	BF%m	BF%m
outlet_r	4.8	15.5	6	61.7	24.7	90.3	34.7	0	0
outlet_l	5.1	15.3	5.3	52.6	21.5	94.4	32.7	0	0
distal	9.9	15.9	5.7	59.1	27.7	193.8	69.5	1	0.32
middle_dist	9.7	16.1	6	79	31.1	181.7	67.1	0.9	0.22
proximal	8.8	18.2	6.3	76.2	28.9	184.5	63.3	0.96	0.2
middle_prox	8.5	18.4	6.3	74.8	26.2	187.1	64	0.15	0.02
inlet	10.7	12.9	4.2	56.5	23.4	172.5	56.4	0.46	0.19

Translate and rotate the plane to inspect local flow and temporal measurements

Backend preprocessing, browser-side interaction



Preprocess once

Update measurements locally

Interactive updates reuse data already loaded in the browser

Scientific representations for the browser

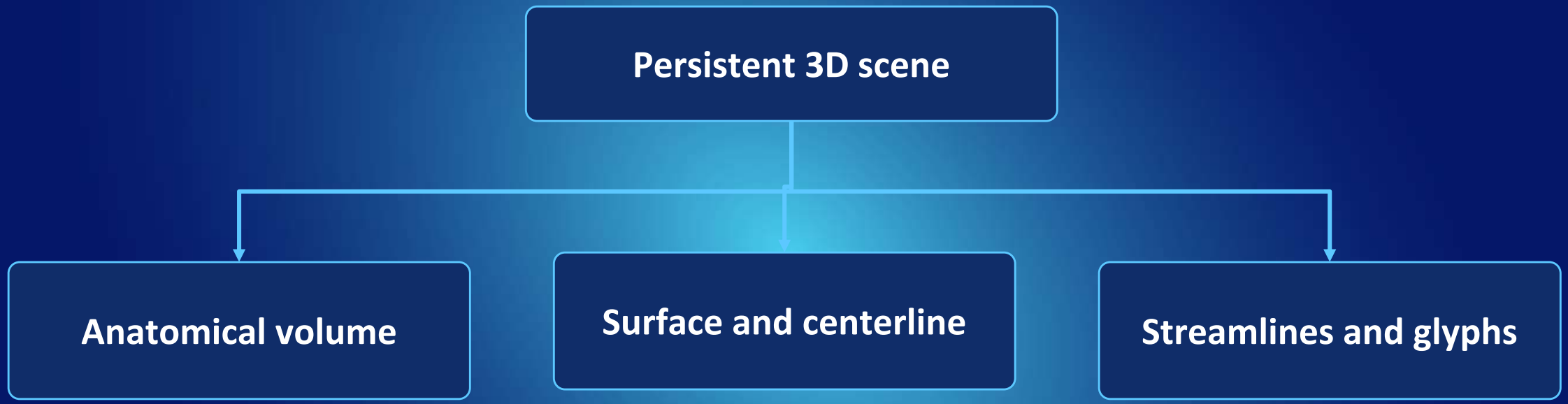
Representation	Purpose
Surface and centerline	Anatomy and initial plane locations
Streamlines	Time-resolved flow visualization
Velocity glyphs and attributes	Local flow sampling and analysis
Anatomical volume	Structural context in the 3D scene
Plane descriptors	Initial analysis positions and orientations

Flat arrays for geometry and scientific attributes. Compressed VTI for anatomy.

The full volumetric tetrahedral velocity field remains on the backend



A persistent VTK.js scene



Plane movement

Update geometry, contours and local measurements

Cardiac frame change

Replace affected temporal buffers and actors

Local analysis without the volumetric mesh

BACKEND REFERENCE

Full volumetric velocity field

↓

PyVista volumetric slicing

BROWSER

Sparse velocity samples

↓

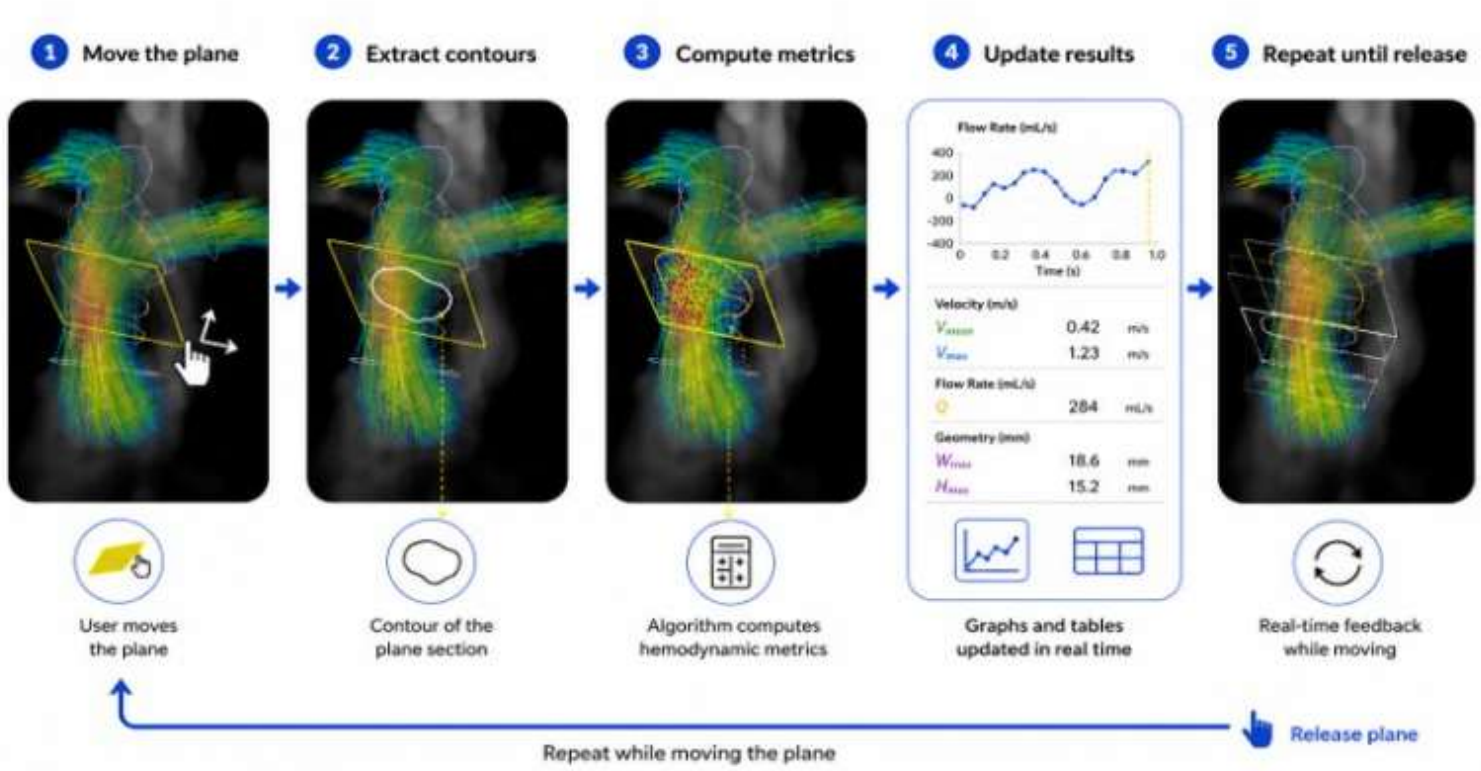
Surface contour + local reconstruction

The contour defines the vessel cross-section

The sampling stage estimates the velocity field inside that contour



Local velocity reconstruction



Metrics update compute flow

SLICING

The VTK.js cutter extracts the vessel contour on the plane section

SAMPLING

Velocity glyphs within ±0.4 cm are interpolated onto a 0.18 cm grid (knn-nearest neighbor weighted interpolation)

KERNEL

Flow rate, velocities, Reynolds number, vorticity and helicity are re-computed

Computation stages and implementation choices



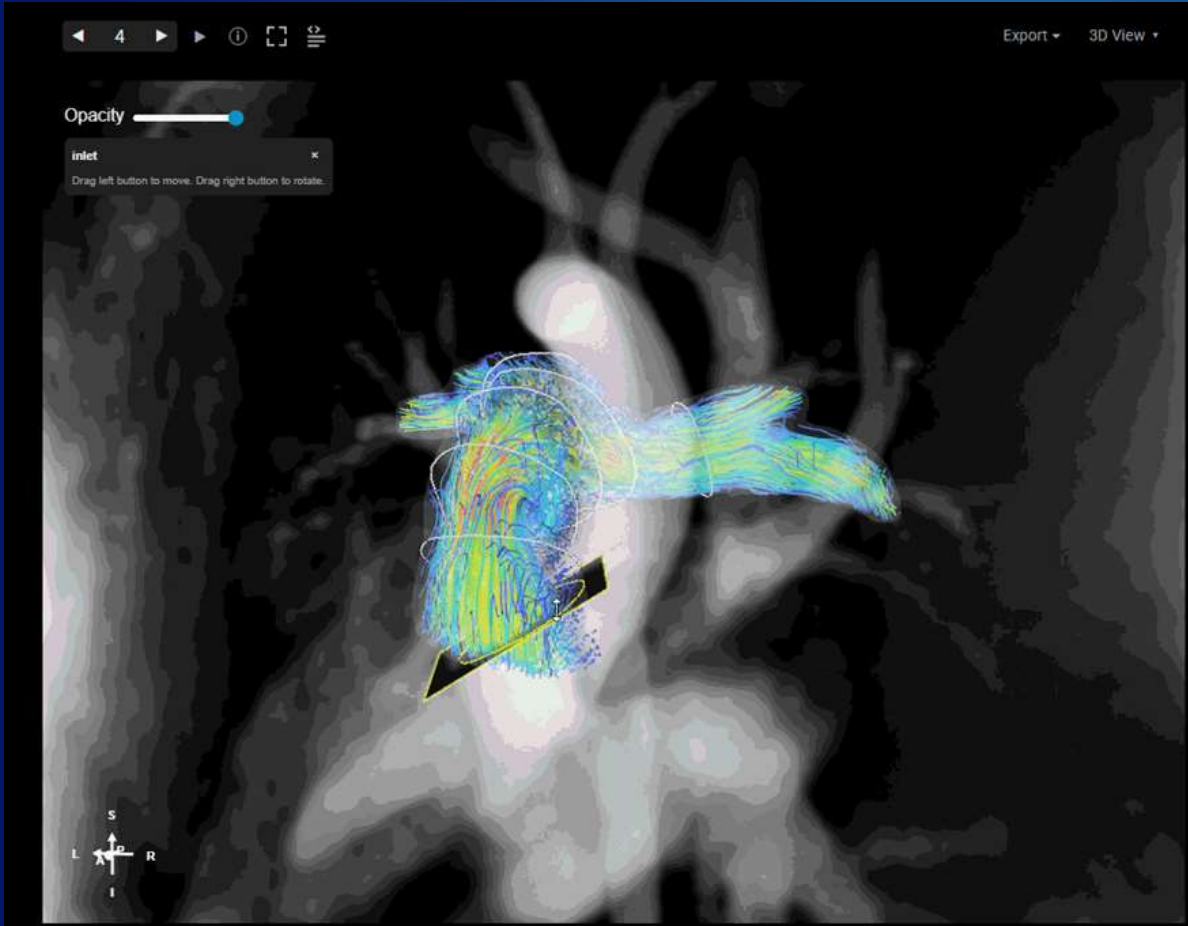
JavaScript established the browser workflow

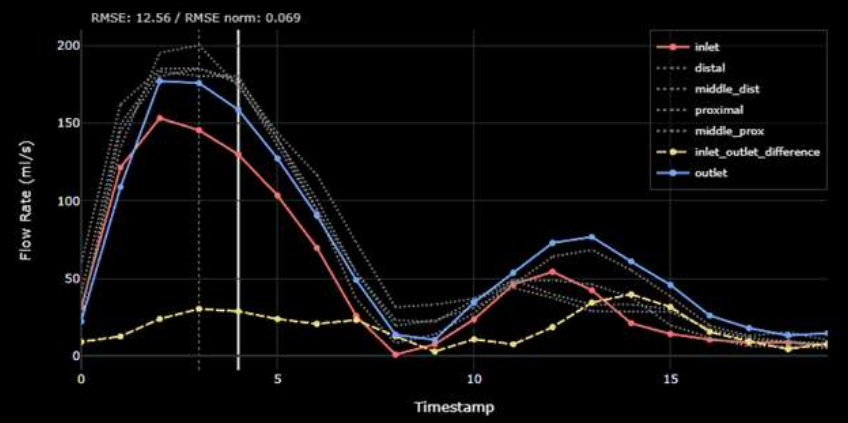
Direct integration with VTK.js, with temporary-allocation overhead

Persistent Rust/Wasm modules reduce numerical execution time

Python alternatives: Pyodide and py2wasm introduce runtime overhead
in the evaluated implementations

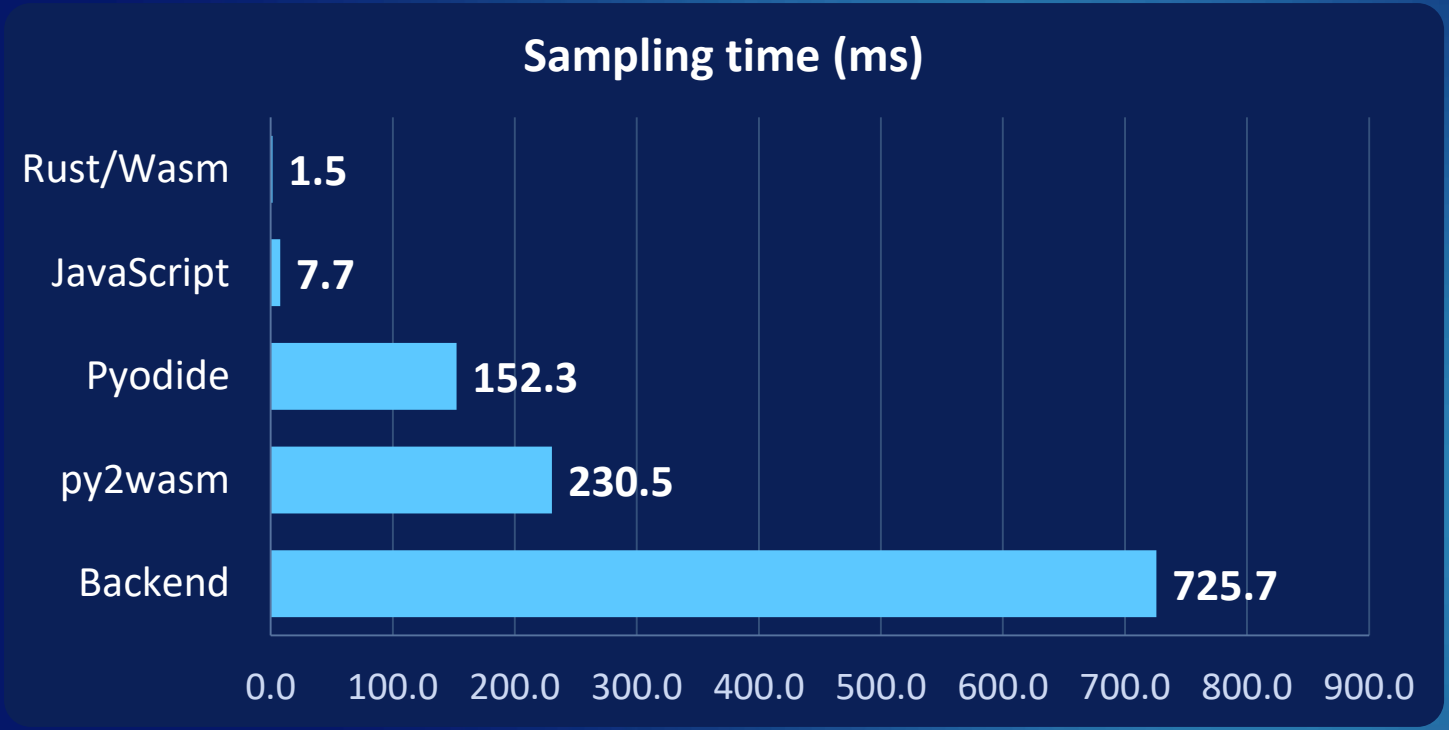
Local velocity reconstruction

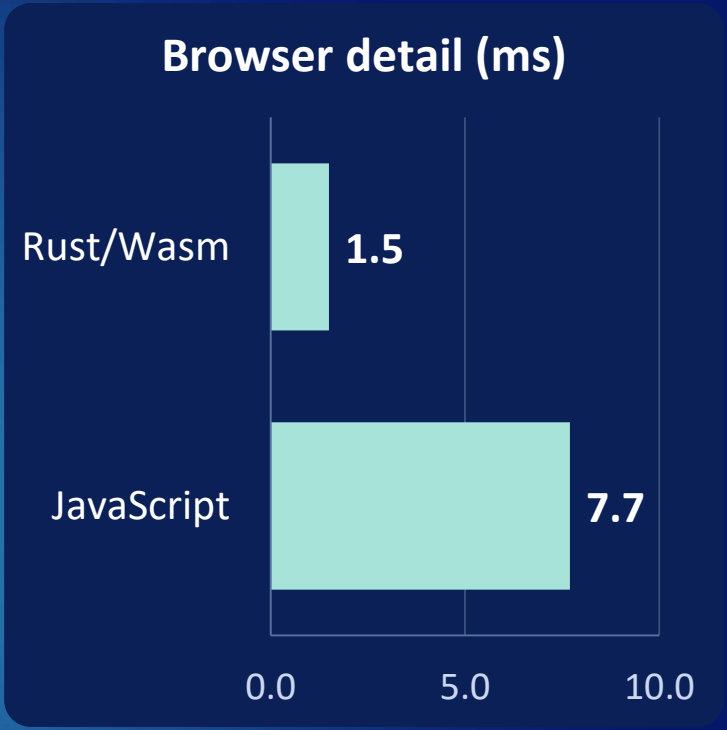




Position	Area	mVmax	mVmean	pVmax	pVmean	FRmax	FRmean	BF%max
inlet	8.6	17.81	5.96	48.91	22.43	153.13	51.23	0.68
outlet_r	5.2	14.5	5.5	51.9	22.3	89.9	34.7	0
outlet_l	5.8	12.9	4.7	56.8	22.6	90.8	32.8	0.4
distal	11.4	14.8	5.3	63.5	28.8	200	71.6	0.77
middle_dist	10.8	15.1	5.6	70	30.2	185.1	69	0.98
proximal	10.4	15.4	5.5	73.5	27.3	183.3	65.7	0.86
middle_prox	9.8	16.3	5.8	75.6	27.1	185.1	65.7	0.56

Execution time during plane interaction



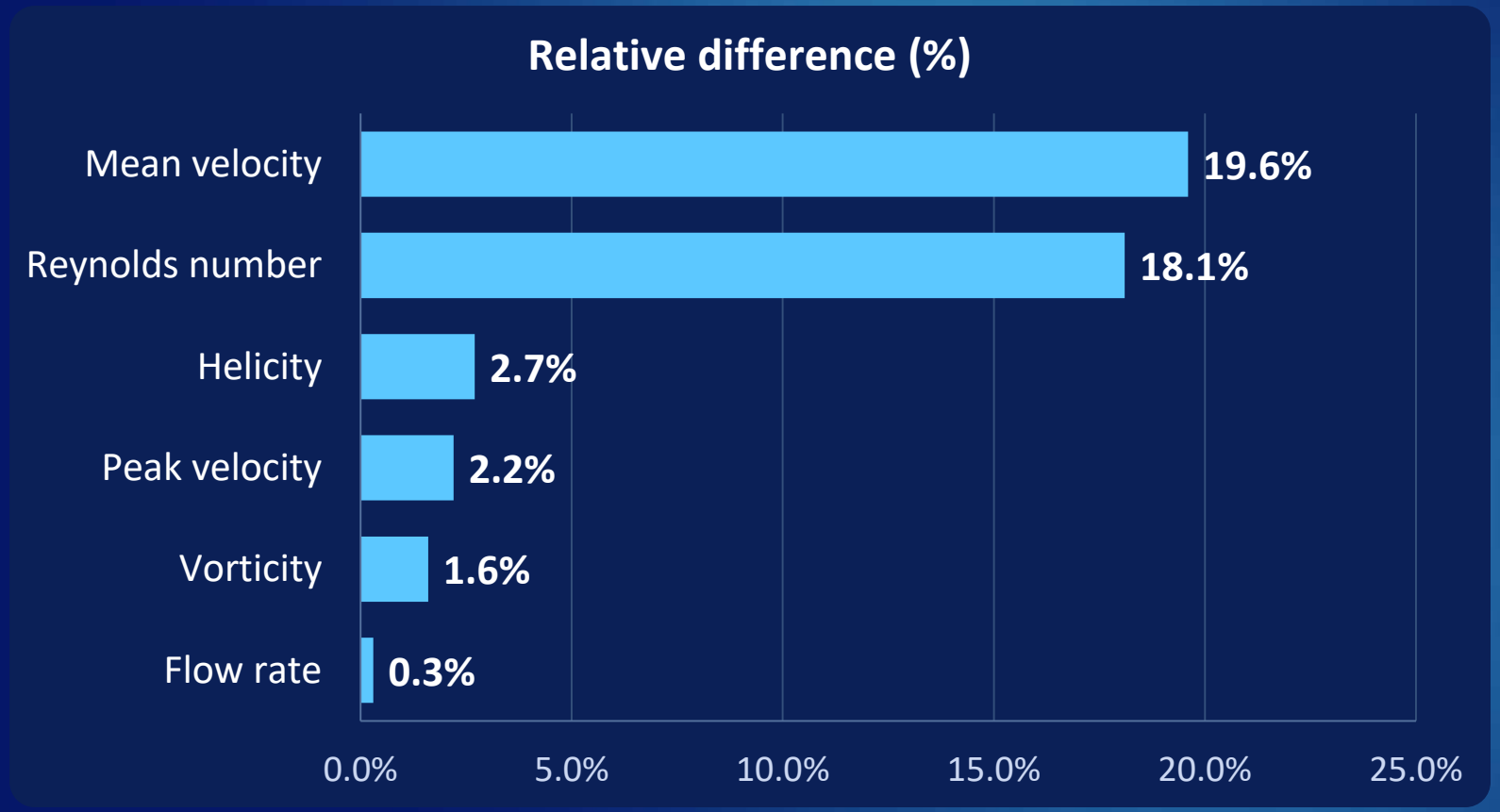


1.5 ms sample + 0.1 ms kernel

JavaScript and Rust finish before the next drag event in the reported tests

Mean stage times. Rendering and network transfer excluded.

Agreement with the backend reference



4 of 6

metrics below 5%

Largest differences: mean velocity and Reynolds number

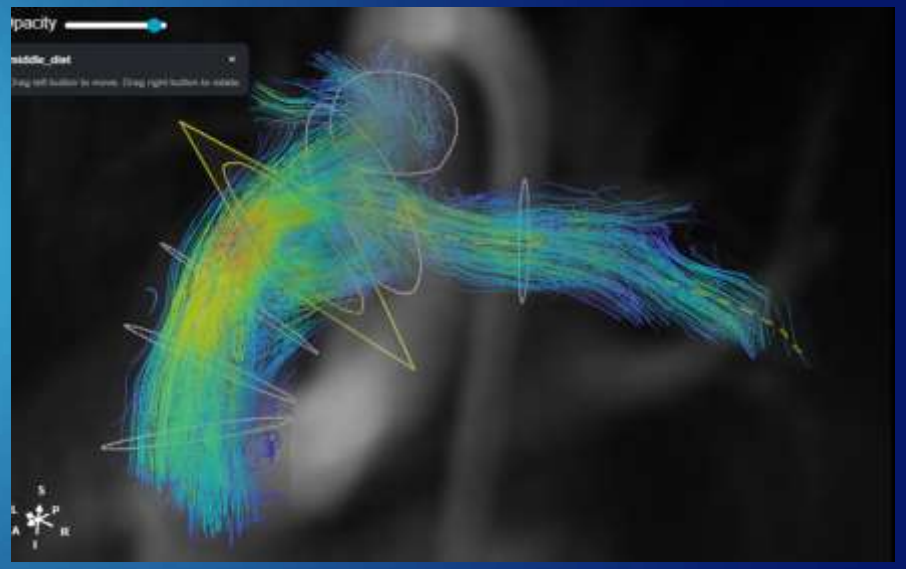
Identical planes and frames. JavaScript and Rust/Wasm give identical results.

Conclusions and next steps

Interactive analysis can run in the browser

Hybrid architecture

Preprocess once and reuse scientific data



Responsive numerical computation

Rust/Wasm accelerates local sampling
and hemodynamic metric evaluation

Next: validate sampling differences and extend multi-plane analysis

Thank you **mcartajena@vicomtech.org**