HiLSVA User Study Document

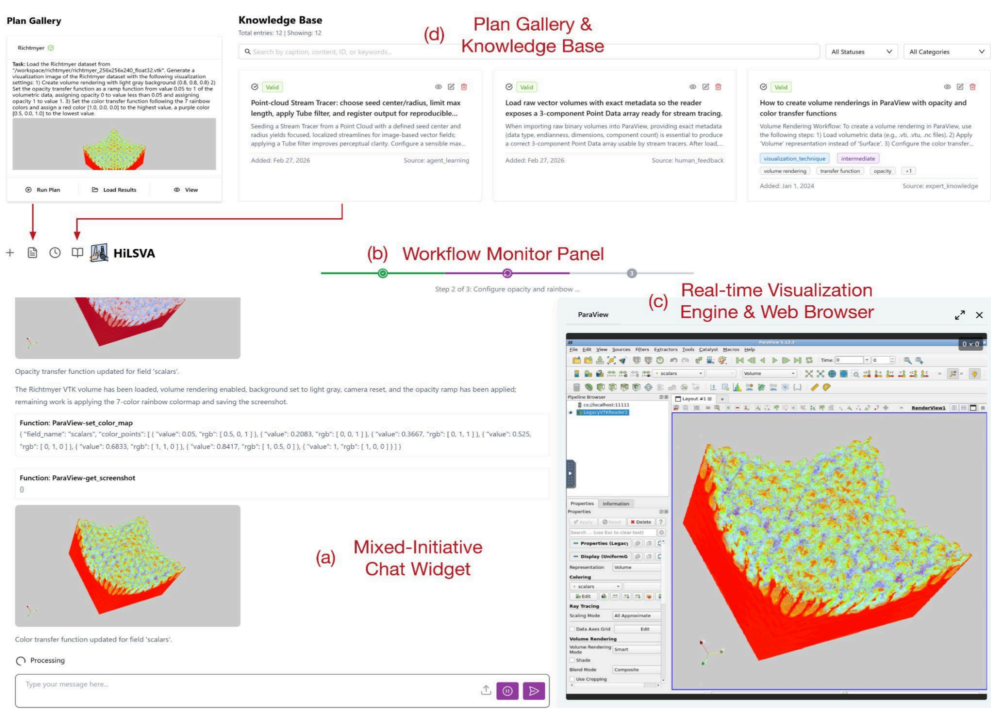
1. What is HiLSVA?

HiLSVA is a human-in-the-loop, agentic system for scientific visualization. It helps you create, modify, and iterate on visualization workflows using natural language while keeping you in control through plan approval, step-by-step execution, and workflow provenance. The backbone visualization engine is ParaView.

What to expect in this study:

- You will work with HiLSVA through a chat interface and a visualization interface.
- You may ask HiLSVA to perform actions, inspect intermediate results, and revise plans.
- Your interactions will be recorded and logged for research purposes.

2. Interface Overview



The HiLSVA interface includes the following main components:

- **Mixed-initiative chat widget:** State goals, review or edit plans, approve actions, and provide feedback.

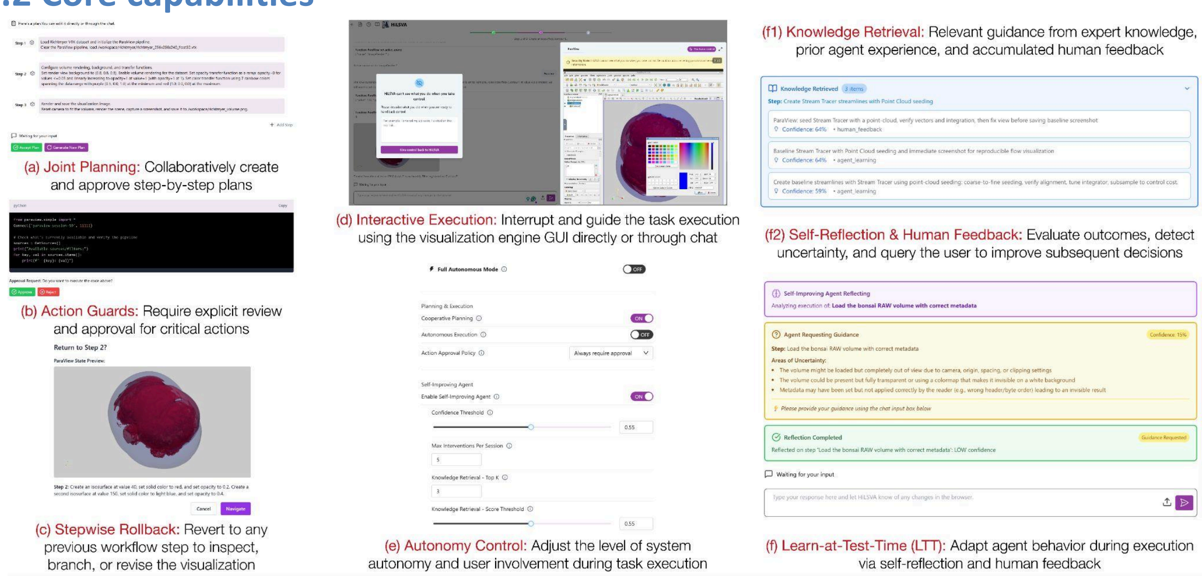
- **Real-time visualization engine & web browser:** View the active ParaView workspace and related tools as they update in real time during agent execution or manual intervention.
- **Workflow monitor panel:** Track step-by-step progress and navigate to earlier steps.
- **Plan gallery:** Revisit previously saved and validated workflows so you can reuse saved plans.
- **Knowledge base:** Browse accumulated guidance, learned feedback, and associated provenance from earlier interactions.
- **Session controls:** Manage active sessions and check whether HiLSVA is running, waiting for approval, or ready for new input.

3. System Usage

3.1 The interaction loop (plan, approve, execute)

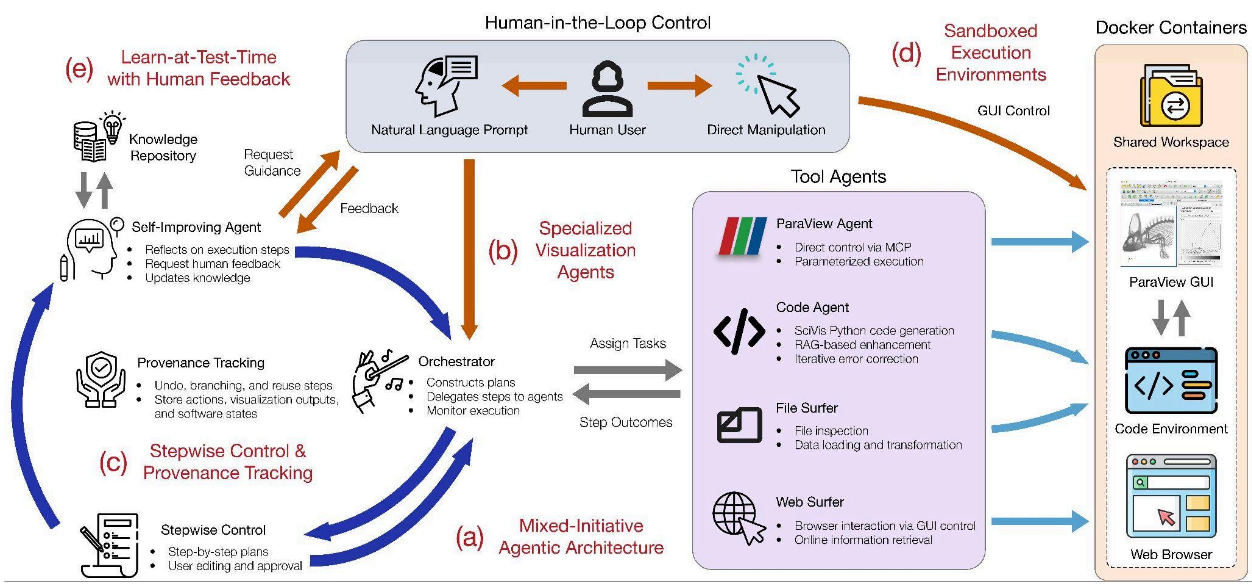
- **Describe your goal**: Tell HiLSVA what you want to achieve and any constraints.
- **Review the proposed plan**: HiLSVA will propose a step-by-step plan. You can edit, reorder, or refine steps.
- **Approve to execute**: Execution starts only after you approve the plan or the next step.
- **Monitor and refine**: During execution, you can pause, ask questions [and resume?], or revise the plan.
- **Rollback when needed**: If something looks wrong or you want to try another approach, revert to an earlier step and try a different choice.

3.2 Core capabilities



- **Joint planning**: collaboratively create and refine a step-by-step workflow.
- **Action guards**: approve critical actions before they happen.
- **Stepwise rollback**: return to a previous step to revise or branch.
- **Interactive execution**: interrupt and guide the system during execution.
- **Autonomy control**: adjust the level of user involvement during task execution.
- **Learning from feedback**: your feedback can help HiLSVA adjust its future behavior within the session.

3.3 What you can ask HiLSVA to do



- **Data and setup:** Load data, inspect metadata, set camera views, and manage sessions.
- **Visualization construction:** Create or modify renderings, add filters, and adjust transfer functions or parameters.
- **Exploration and iteration:** Try alternatives, compare results, and branch from earlier steps.
- **Web search and external reference gathering:** Search the web for relevant documentation, images, or domain background that can support the current visualization task.
- **Knowledge base support and reuse:** Access prior workflows, saved plans, and learned guidance from the knowledge base.
- **Export and reproducibility:** Save images and record steps so the workflow can be revisited.

3.4 Safety and control notes

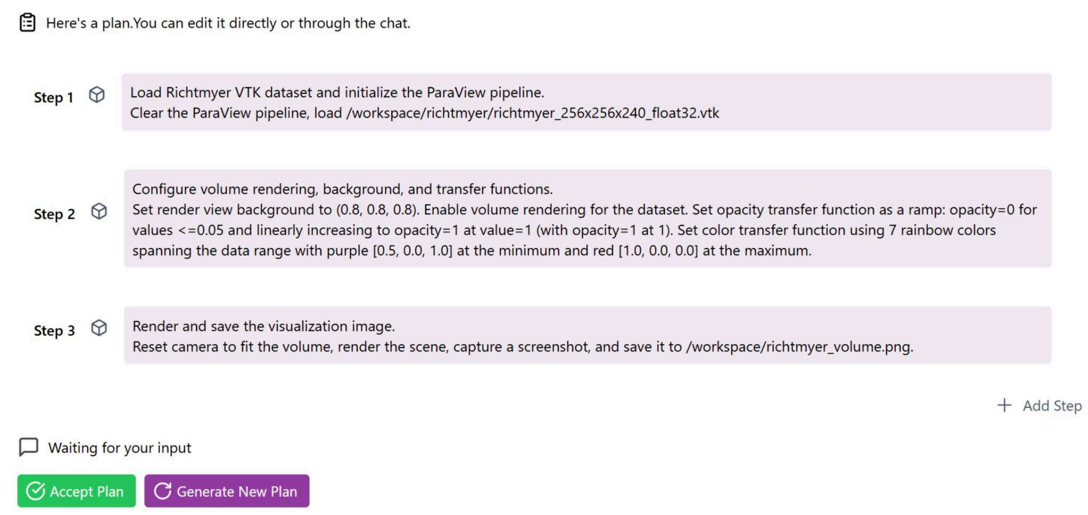
- HiLSVA requests approval before performing critical actions (can be turned off).
- You can stop execution and take manual control at any time.

4. Example Interactions

Example 1: Plan request, edit, approve

User: “Load the dataset and apply a volume rendering. “

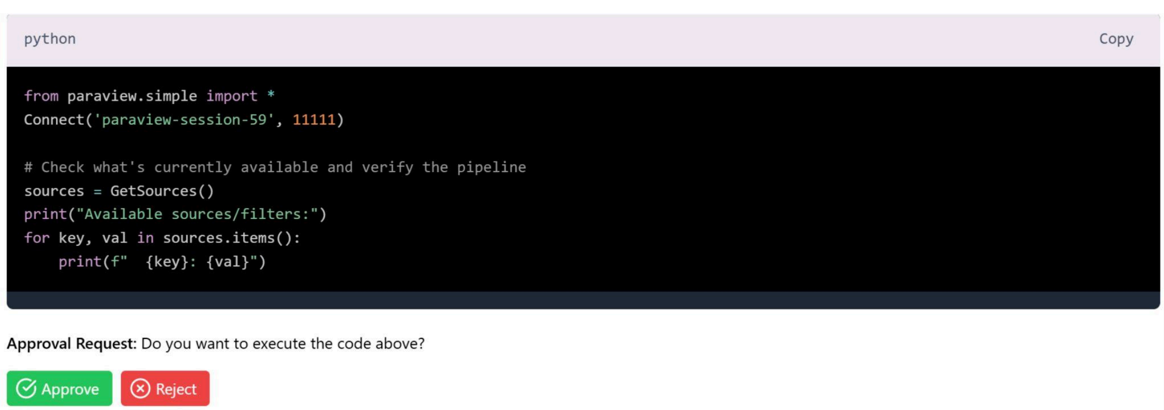
HiLSVA: Proposes a multi-step plan. Users can edit and approve the plan.



Example 2: Guarded action

User: “Generate and run a code script.”

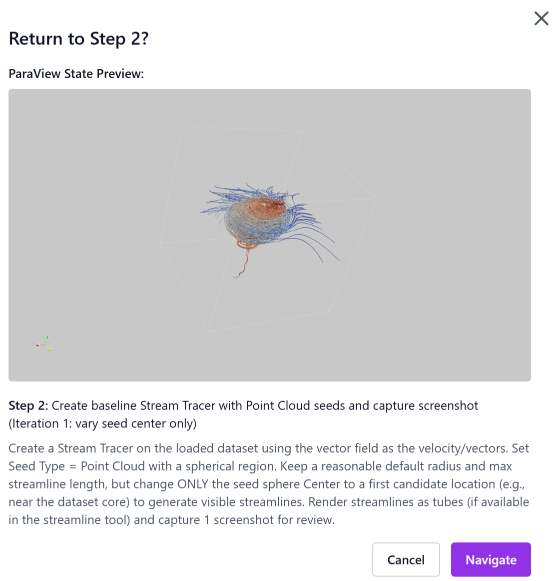
HiLSVA: Shows what will be executed and asks for approval. You can turn it off in the autonomy setting page.



Example 3: Rollback and branch

User: "Go back to Step 2 and try a different threshold, keeping everything else the same."

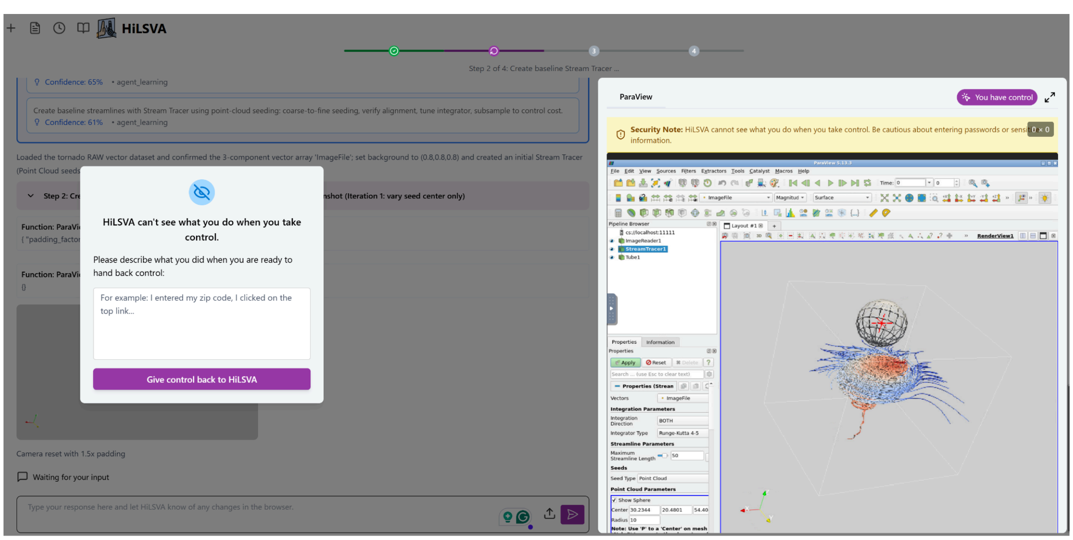
HiLSVA: Restores the earlier state and applies the change to produce an alternative result. All the ParaView states will be restored to that step.



Example 4: Direct GUI intervention

User: "I will adjust the transfer function manually."

User's Action: Make manual adjustments in the visualization workspace.



5. Your Tasks

5.1 Introduction stage (5 minutes)

- A brief overview of HiLSVA using the Richtmyer dataset.
- Watch the demonstration video showcasing the interface and key features.

5.2 Exploration stage (10 minutes)

- You will have time to explore the system on the Richtmyer dataset.
- Load the Richtmyer dataset from "/workspace/richtmyer/richtmyer_256x256x240_float32.vtk".
- You are encouraged to ask questions and get familiar with the interface.

5.3 Task stage (35 minutes)

At this stage, your interactions will be recorded for research analysis.

During the study, you will complete **four visualization tasks**.

HiLSVA supports different **levels of autonomy**, which control how actively the system performs actions versus how much user guidance is required.

For the **first three cases (Hurricane, Foot, Tornado)**, the researchers will decide, for each task, which one of the three autonomy settings the participant will use:

1. **Full-autonomous mode**
   Turn **on** "Full Autonomous Mode."

2. **Half-autonomous mode without learning at test time (LTT)**
   Turn **off** "Full Autonomous Mode" and "Enable Self-Improving Agent."

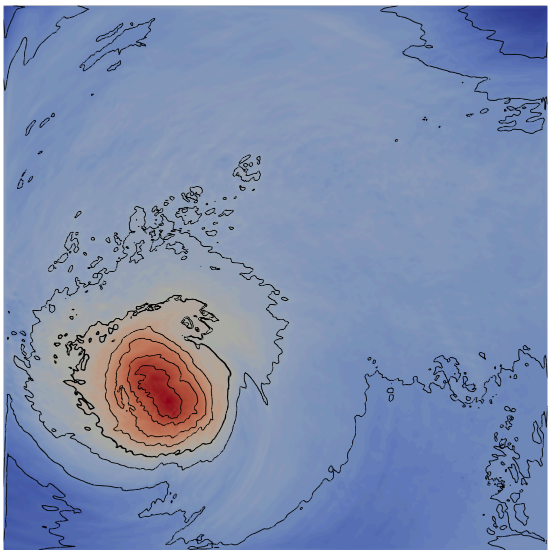
3. **Mixed-initiative mode with LTT**
   Turn **on** "Enable Self-Improving Agent" and use the following settings:
   - Max Interventions: 5
   - Knowledge Retrieval – Top K: 3
   - Knowledge Retrieval – Score Threshold: 0.55
   - Confidence Threshold: 0.65

For the **fourth task (combustion)**, you may **freely choose whichever autonomy setting you prefer**.

Case 1: Hurricane

This dataset is a simulation of Hurricane Isabel with spatial resolution 500 × 500 × 100 (float32, little endian).

In this task, you will visualize the temperature field to inspect the hurricane structure.



Task 1 — Create a slice visualization

Load the Hurricane dataset from "/workspace/isabel/isabel_500x500x100_float32.raw" and create a **horizontal slice along the z-axis** at z=49.5.

Apply a preset "Color to Warm" colormap to visualize the temperature field.

Save a screenshot of the result to "/workspace/P{your_ID}/isabel_slice.png".

Task 2 — Inspect the value range

Create a histogram of the slice data to examine the distribution of temperature values.

Record the value range (minimum and maximum values) of the slice.

Q1: What is the value range of the slice?

Your answer:

Task 3 — Improve the visualization

Refine the visualization by adding contour lines over the slice.

Generate 10 contour lines by linearly sampling isovalues within the value range obtained in Task 2. This should help reveal the hurricane structure more clearly.

Save the refined visualization to “/workspace/P{your_ID}/isabel_final.png”.

Task 4 — Data interpretation

Q2: What is the purpose of adding contour lines to the slice?
A. To smooth the dataset
B. To highlight equal-value structures and reveal patterns
C. To reduce the dataset size
D. To convert the data into vectors
Your answer:

Q3: Based on the visualization, which region of the hurricane generally has higher temperatures?
A. The hurricane eye
B. The surrounding spiral bands
C. The outer atmosphere
D. All regions have the same temperature
Your answer:

Case 2: Foot

This dataset is a **CT scan of a human foot** with spatial resolution 125 × 255 × 183 (float32, little endian). The data contains different tissues such as **bone, skin, and soft tissue**.

In this task, you will use **isosurface rendering** to reveal the internal bone structures.

Task 1 — Generate an isosurface visualization

Load the Foot dataset from “/workspace/foot/foot_125x255x183_float32.raw”. Use isosurface rendering to separate bone structures from surrounding soft tissue. Render bones in white, and skin/soft tissue in warm orange.

You may adjust the isosurface value(s) to obtain a visualization where the bones of the foot are clearly visible.

Save a screenshot of your visualization to “/workspace/P{your_ID}/foot_isosurface.png”.

Task 2 — Retrieve an anatomy reference

Ask HiLSVA to find a labeled human foot anatomy image online that shows key bone structures such as the metatarsals and phalanges. Briefly inspect the reference image before continuing.

Task 3 — Refine the visualization

Using the anatomy reference as guidance, further adjust the isosurface value(s) and visualization settings until the major bones of the foot are clearly visible.

You may change: isosurface value(s), color, and camera view

Save the refined visualization to “/workspace/P{your_ID}/foot_final.png”.

Task 4 — Anatomical interpretation

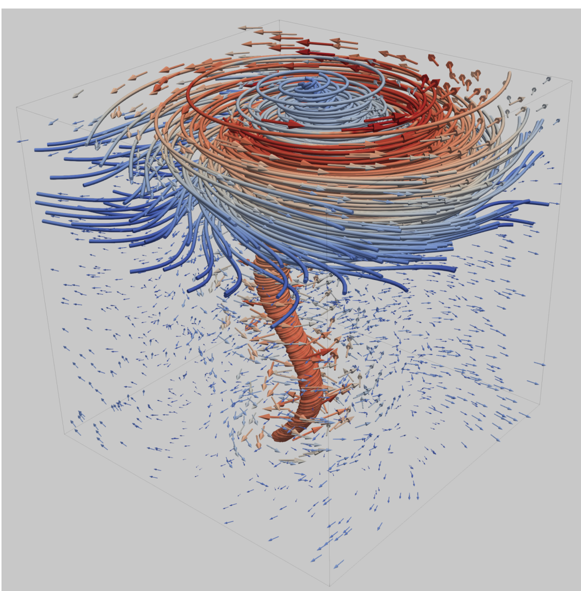
Inspect the visualization you created and answer the following questions.

Q1: Based on the isosurface rendering of the foot dataset, which of the following best describes the visibility of bony structures?
A. Both the phalanges and metatarsals are fully visible
B. The phalanges are fully visible, but the metatarsals are only partially visible
C. The metatarsals are fully visible, but the phalanges are only partially visible
D. Neither the phalanges nor the metatarsals are clearly visible
Your answer:

Q2: What is the main purpose of adjusting the isosurface value in this task?
A. To change the file format of the dataset
B. To control which tissue density is visualized
C. To reduce the dataset size
D. To convert the dataset into a vector field
Your answer:

Q3: Which anatomical structures correspond to the long bones in the middle of the foot?
A. Tarsals
B. Tibia
C. Phalanges
D. Metatarsals
Your answer:

Case 3: Tornado



This dataset is a simulated tornado vector field with spatial resolution 64 x 64 x 64. Each voxel stores a 3D velocity vector that represents the airflow in the tornado. In this task, you will visualize the flow structure using streamlines and glyphs.

Task 1 — Generate initial streamlines

Load the Tornado dataset from "/workspace/tornado/tornado_64x64x64_float32_scalar3.raw". Use the Stream Tracer filter with Point Cloud seeding to generate streamlines.

Perform three visualization trials (each step one trial), varying only one parameter per trial:

Trial 1

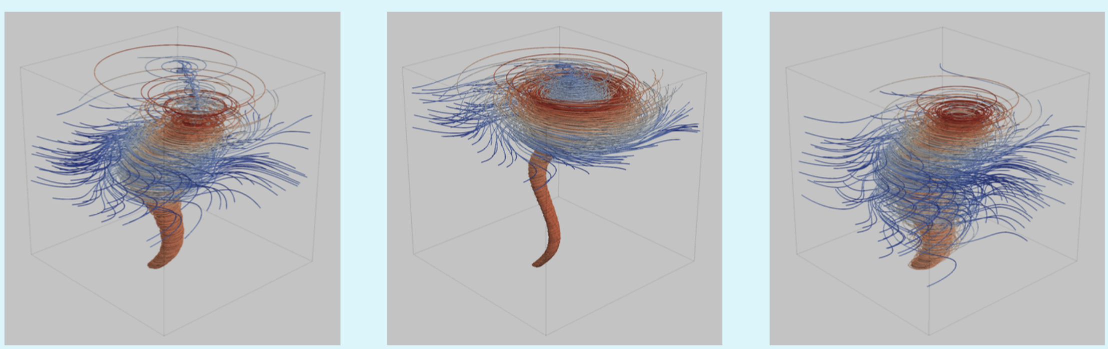
- Seeding sphere center: [31.5, 31.5, 31.5] (center of the volume)
- Seeding sphere radius: 10
- Maximum streamline length: 512

Trial 2: Change only the seeding sphere center.

- Seeding sphere center: [31.5, 31.5, 47.5]
- Seeding sphere radius: 10
- Maximum streamline length: 512

Trial 3: Reset the center and change only the seeding sphere radius.

- Seeding sphere center: [31.5, 31.5, 31.5]
- Seeding sphere radius: 20
- Maximum streamline length: 512



Trial 1: center [31.5, 31.5, 31.5], radius 10, max length 512.

Trial 2: center [31.5, 31.5, 47.5], radius 10, max length 512.

Trial 3: center [31.5, 31.5, 31.5], radius 20, max length 512.

Task 2 — Select the best visualization

Inspect the three streamline visualizations and determine which one best reveals the tornado flow structure. Select the preferred trial. Use HiLSVA to restore the visualization corresponding to that trial.

Task 3 — Add vector glyph visualization

Add a Glyph filter on the original vector field using arrow glyphs. Orient arrows by the Velocity vector and scale by Velocity magnitude with a scale factor of 25.0. Set the maximum number of sample points to 2500.

Task 4 — Refine the visualization

Improve the clarity of the streamline visualization by adjusting parameters such as tube radius. Save the final visualization to "/workspace/P{your_ID}/tornado_final.png".

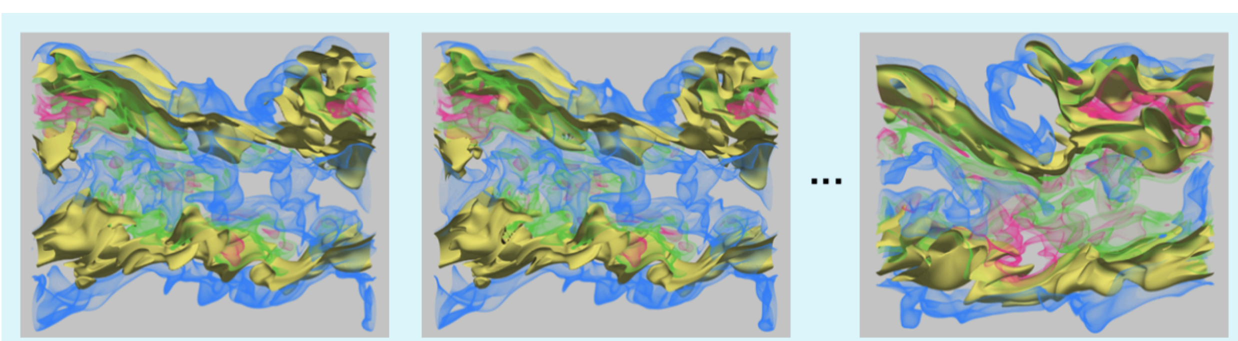
Task 5 — Data interpretation

Q1: What is the purpose of using streamlines in this dataset?
A. To display temperature values
B. To visualize the direction and structure of the flow field
C. To reduce the size of the dataset
D. To convert vectors into scalar values
Your answer:

Q2: In the glyph visualization, what does the orientation of each arrow represent?
A. Camera direction
B. Grid resolution
C. Velocity vector direction
D. Color mapping
Your answer:

Q3: What does the size of each arrow glyph represent in this task?
A. Distance from the center
B. Time step
C. Dataset resolution
D. Velocity magnitude
Your answer:

Case 4: Combustion



This dataset is a multivariate, time-varying simulation of turbulent reacting flow. It contains multiple timesteps with a spatial resolution of 240 x 360 x 60.

Two scalar variables are used in this task:

- Y_OH — the mass fraction of the hydroxyl radical, commonly used as an indicator of reaction zones in combustion
- mixfrac — the mixture fraction of fuel and oxidizer

In this task, you will visualize the time-varying Y_OH field using volume rendering and export an animation. Then you will analyze another variable (mixfrac) using isosurface visualization.

Task 1 — Determine the number of timesteps

Inspect the dataset directory "/workspace/combustion/jet_Y_OH/", the directory contains files named:

├── jet_Y_OH_240x360x60_float32_0000.raw

├── jet_Y_OH_240x360x60_float32_0001.raw

├── …

Q1: What is the **number of timesteps** in the dataset?

Your answer:

Task 2 — Visualize the Y_OH variable

Load the time-varying Y_OH variable from "/workspace/combustion/jet_Y_OH/jet_Y_OH_240x360x60_float32_0000.raw".

Perform **volume rendering** using the given Y_OH colormap in "/workspace/combustion/Y_OH.json".

Export the animation as a series of images to "/workspace/P{your_ID}/Y_OH_dvr/timestep_xxxx.png".

Task 3 — Visualize another variable

Load the time-varying mixfrac variable from "/workspace/combustion/jet_mixfrac/jet_mixfrac_240x360x60_float32_0000.raw".

Extract an **isosurface with isovalue = 0.4 and yellow color**. Generate the isosurface visualization across all timesteps.

Export the animation as a series of images to
"/workspace/P{your_ID}/Y_OH_dvr_mixfrac_iso/timestep_xxxx.png".

Task 4 — Data interpretation

Q2: After generating the animation of the **Y_OH** field, which of the following best describes what you observe?
A. The visualization remains identical across all frames
B. The dataset only changes color but not structure
C. The reaction structures move and change shape over time
D. Only the camera view changes
Your answer:

Q3: After generating the **mixfrac isosurface (value = 0.4)**, what does the extracted isosurface represent?
A. Regions where the mixture fraction equals the selected value
B. Regions with the highest velocity magnitude
C. Regions where the combustion process stops
D. Regions where the dataset has missing values
Your answer:

6. Post-Study Questionnaire (10 minutes)

After completing the tasks, you will answer a short questionnaire about your experience:

Part 1: Participant Background

1. What is your field of study? (Open-ended response)

2. How would you rate your experience with visualization tools (in particular, ParaView or VTK)?
◯ No experience
◯ Some experience
◯ Expert

3. How familiar are you with interactive visualization systems?
◯ Not familiar
◯ Somewhat familiar
◯ Very familiar

4. How familiar are you with natural language-driven visualization systems?

◯ Not familiar
◯ Somewhat familiar
◯ Very familiar

5. How well do you know the datasets used in this experiment? (foot, hurricane, tornado, combustion)
◯ No prior knowledge
◯ Basic knowledge
◯ Extensive knowledge

Part 2: System Usability Evaluation

Use the same 5-point Likert scale:
1 = Strongly Disagree, 2 = Disagree, 3 = Neutral, 4 = Agree, 5 = Strongly Agree

1. HiLSVA helped me complete scientific visualization tasks efficiently.
2. I can easily complete visualization tasks without assistance.
3. The system's functions are well integrated and work seamlessly together.
4. The system is easy to learn, even for first-time users.
5. The step-by-step plan made it clear what the system intended to do before execution.
6. I felt in control of the workflow because I could review, edit, or approve actions before they were carried out.
7. I felt that I could easily adjust the balance between system automation and my own control.
8. The system provided clear feedback about what it was doing at each step.
9. The record of previous steps helped me understand and trust the system's behavior.
10. It was easy to go back to an earlier step and try a different approach.
11. The combination of natural-language interaction and direct visualization interaction worked well together.
12. The system was able to recognize when it was uncertain and ask for my guidance.
13. I felt that the system benefited from accumulated knowledge or past interactions.
14. The system's interface is intuitive and user-friendly.

Part 3: Open-Ended Feedback

1. What aspects of the system did you find most useful? (Open-ended response)
2. What challenges did you face while using the system? (Open-ended response)
3. What improvements or additional features would you suggest for HiLSVA? (Open-ended response)