

# FIELDVIEW

## What's new in FieldView 12.2

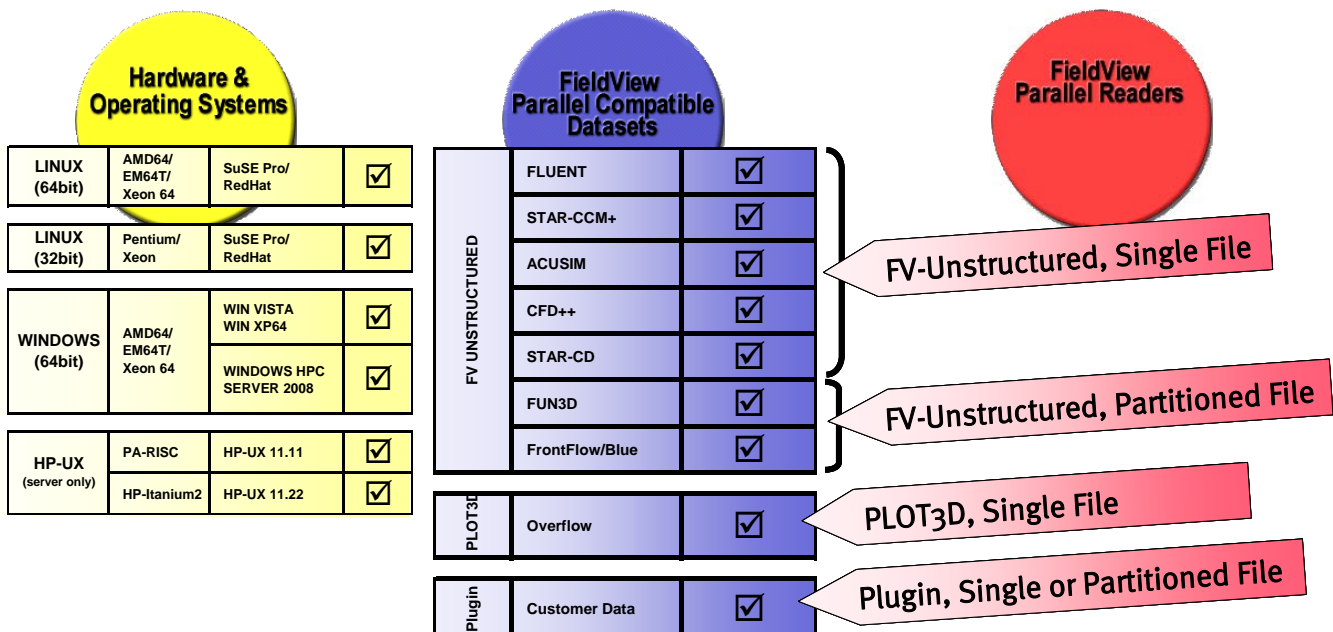
FieldView 12.2 is a point release, designed to bring considerable gains to your CFD workflows! We have greatly advanced the parallel processing features in FieldView - ease of use AND high-performance, all in one!

### SUMMARY OF NEW FEATURES

- Realize performance benefits with FieldView Parallel for WINDOWS
- Speed up access to your in-house data using Plugin Parallel Readers
- High performance Partitioned File Parallel support more data types
- Run FieldView Parallel at no additional cost on up to three processes
- Easily integrate & access your FVX programs directly from FieldView GUI
- Control & automate your workflows with new FVX and SCRIPT commands
- Extend your ability to explore 2D datasets with 2D to 3D Dataset Extrusion
- Make faster streamlines & cleaner cut planes with Improved Data Handling
- New mouse controls on WINDOWS
- Read UH3D Data on WINDOWS
- Read larger files w/ MAC 64bit server
- Shading improved for multigrid data

### FieldView Parallel

This release contains the completed delivery of FieldView Parallel. This fulfillment covers three areas, namely: Hardware & Operating Systems, Parallel Compatible Datasets and Parallel Readers. Adding

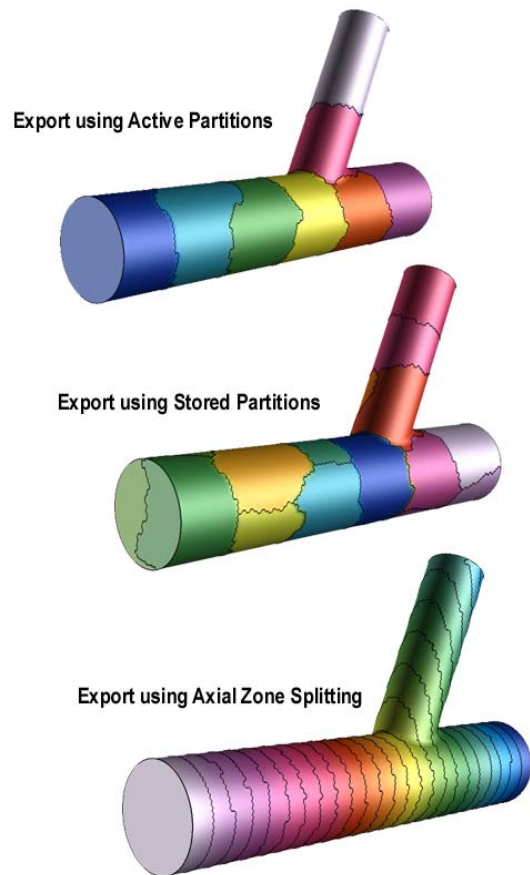


to our full coverage of 64bit and 32bit LINUX systems, and our support for HP-UX platforms, we now run on 64bit WINDOWS including the latest WINDOWS HPC Server 2008. Specific details for certified and supported platforms are available on our website.

Second, most commercial solvers are now able to create FieldView “parallel compatible” exports. A new FLUENT scheme file, distributed by Intelligent Light, greatly simplifies the creation of multigrid exports. In the picture at right, a FLUENT dataset has been exported in three different ways using this scheme file. Other solvers such as starccm+ and STAR-CD (CD-adapco), AcuSolve (ACUSIM), CFD++ (Metacomp Technologies) are all capable of producing single file multigrid exports. And, the FUN3D and FrontFlow/Blue solver codes export our unstructured partitioned file format, thereby realizing the maximum benefits of FieldView Parallel.

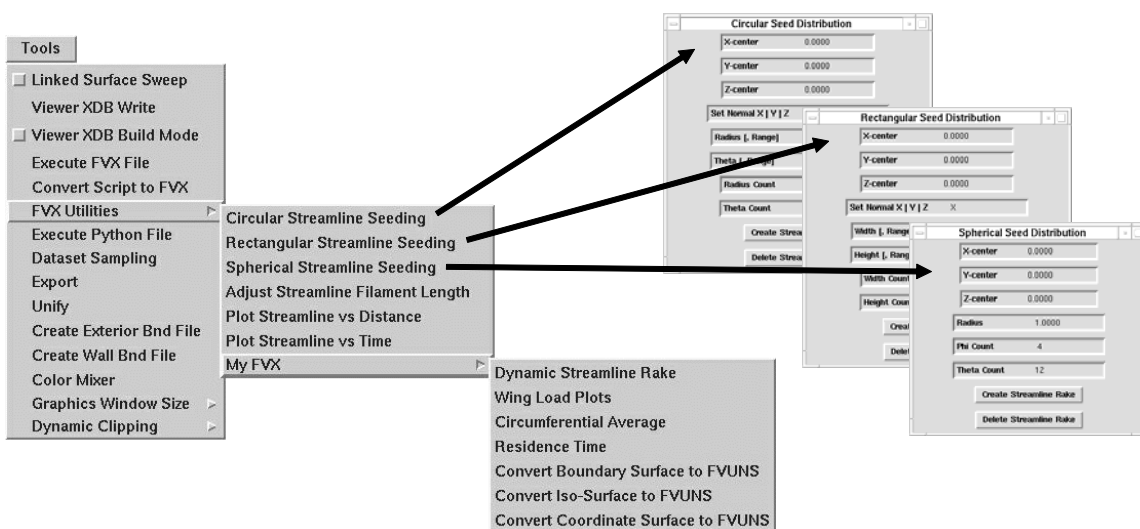
Third, we are delivering **single file multigrid** and **partitioned file parallel** readers for unstructured, structured and PLOT3D data formats. And, for those of you running in-house solvers, the plugin toolkit reader capability can now be used to read either single file multigrid or partitioned file datasets. If you already have developed your own plugin readers, you will be pleased to know that parallel support will already be enabled, and only small changes to register your readers will be needed.

To make FieldView Parallel easily accessible, we allow for the use of **three free processes** at no additional charge. With most cases, you should expect to see a speed-up of up to 2X on operations such as reading data and creating and sweeping surfaces. If you are running FieldView Parallel on a local system, it can be set up to run with a one-line configuration file.



## FVX Improvements

A new entry on the Tools menu, **FVX Utilities**, provides access to the FVX tools and routines which are included with each FieldView release. Some of these routines (illustrated below) are used to spec-

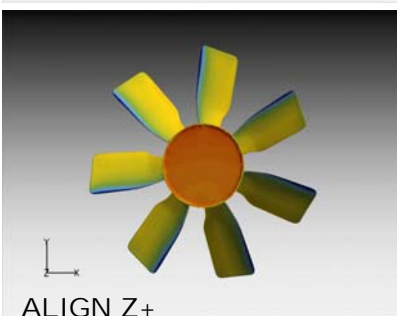
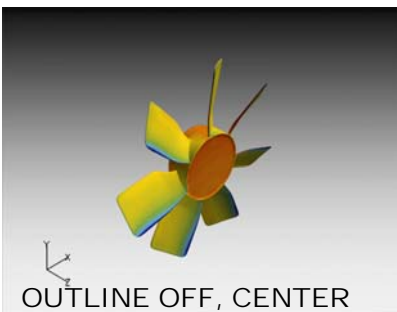
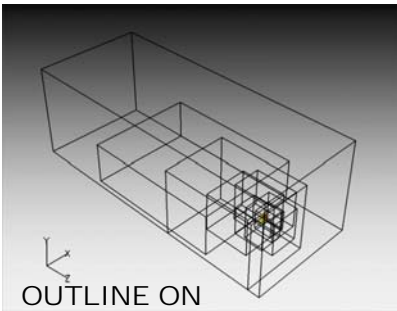


ify the seeding patterns for the creation of streamline rakes. This feature is designed so it can be further customized by you to access any FVX tools that you have developed by following the **My FVX** menu entry .

In prior releases of FieldView, if a surface, rake or annotation object was created interactively, or from a RESTART or SCRIPT, this object was inaccessible to FVX - it could not be modified, queried or deleted. Two new FVX commands resolve this limitation by creating “handles” that can be used by FVX for complete control. One command, `get_current_object_handle`, returns the handle for the “current” object. The other command, `get_all_object_handles`, returns a table with object handles for all surfaces, rakes and annotation objects for the specified dataset. We believe that this feature will make the task of fully automating existing SCRIPTS and RESTARTS much easier since it will no longer be necessary to create objects using FVX commands.

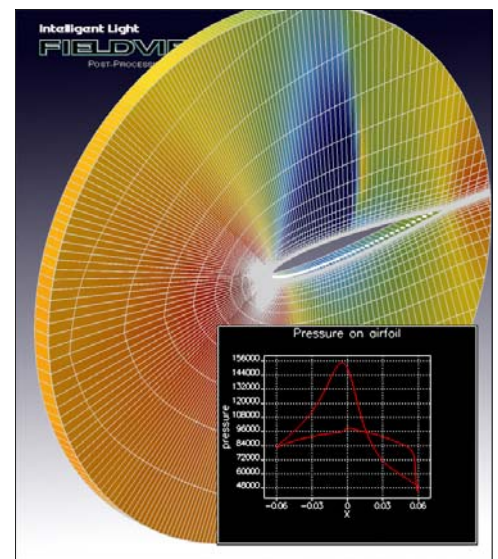
## SCRIPT VIEW Controls

All of the standard view settings such as Center, Reset, Align and Perspective, along with the ability to toggle the settings for the outline and axis marker visibility can now be controlled with SCRIPT commands. This lets you create images of your data automatically without having to first save View RESTARTS. In the illustration at left, a dataset has been read in with the outline ON. Within this picture a boundary surface of a fan is illustrated but it is difficult to see. By first turning the outline OFF and re-centering the data using the CENTER command, the fan blade is now more easily seen. Finally, the fan blade can be aligned normal to the screen by using one of the preset aligned views. Combining these new View controls with other SCRIPT commands such as PRINT will let you fully automate the generation of images for your reports.



## 2D to 3D Flat Surface Extrusion

Previous versions of FieldView have been unable to create 2D plots or to point probe on 2D datasets. In order to overcome this problem, it is now possible to extrude any 2D dataset in a direction normal to the plane of data. The extrusion will work on all binary FieldView unstructured data, including FLUENT, CFD++ and NPARC/WIND solvers. Note that extrusion thickness in the illustration at right is exaggerated and can be controlled using Dataset Scaling.

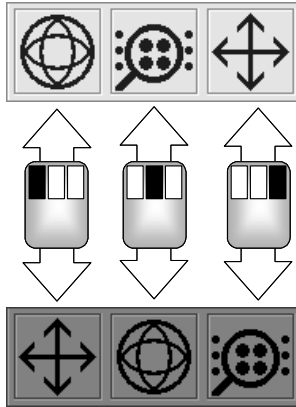


## Improved Data Handling

Changes have been made to interpolation for arbitrary polyhedral cells. As a result, we expect that problems such as holes in coordinate planes, particularly for datasets having hexcore style meshes will be properly resolved. This change in interpolation should also bring improvements to streamlines in the placement of seeds and the calculation of their trajectories. Another fix for derivatives (gradient, divergence and curl) for the special case of split

planar faces is expected in part to resolve the problem of streamlines stopping prematurely within the model domain.

## FieldView for WINDOWS Users



**New Mouse Controls**  
Enabled using -fc switch

**Existing Mouse Controls**  
Maintained as default

**UH3D Direct Reader for WIN64**  
Enables access to large cases



## Other Important Improvements

- If you have experienced **unexpectedly long times** for **streamline calculations**, and you have seen the diagnostic message in the FieldView console window that the maximum 10000 step limit has been reached for a particular streamline seed, then it is likely that this release of FieldView will resolve this problem
- The 64bit SERVER is now the default for the **Apple MAC INTEL platform** running OS X 10.5.x (Leopard) and above.
- A problem with **shading** on an **unstructured boundary surface** when a dynamic clip was applied has been fixed. A related problem where a shading discontinuity had an effect on integration has also been fixed.
- The **FVX delete** command has now been implemented for **user defined GUI panels**.
- A difference between FieldView and FieldView parallel for the specific case of cell orientation on a boundary surface has been resolved.
- Several small fixes for **FieldView Parallel Partitioned file** reading of unstructured datasets have been made.
- A serious problem when changing the **vector display to uniform sampling**, while running FieldView Parallel, has been fixed.
- A problem when reading a split grid & results case, which successfully read into an earlier FieldView release has been resolved.
- **Default file permissions** have been **changed** on the UNIX and LINUX installations to make them more secure.
- The transient optimization was inadvertently broken in the FieldView 12.1 release and has now been fixed. This will speed up unsteady postprocessing.
- An assert with the msg **EFAULT: Bad address**. The system detected an invalid pointer address in attempting to use a pointer argument of a call has been fixed.
- An assert failure associated with **fetch\_element** when attempting to **generate vortex cores** has been fixed.