What's New in AVS/Express 8.0

Licensing updates

New licenses required for 8.0

This release of AVS/Express requires updated licenses. If you currently have a license for 7.0 - 7.3.1, you must contact AVS Support to obtain an updated license file. For compatibility, your new license will work with previous releases.

Support for IPv6

The AVS/Express node-locked and floating licensing now supports HOSTIDs using IPv6 address and address ranges. For more information, see the Detailed Licensing Information appendix of the Installing AVS/Express book.

Platform updates

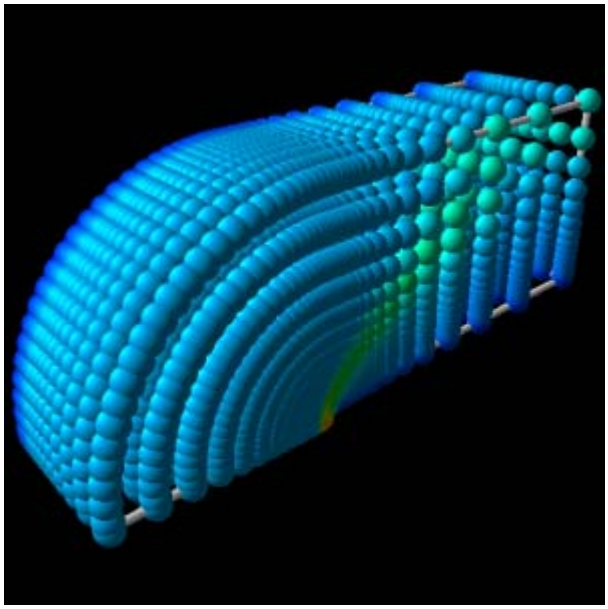
- Support for RHEL6 on x86 and x64-64 platforms.
- The following platforms have been dropped for this release: SGI IRIX64 and HP-UX on IA64.

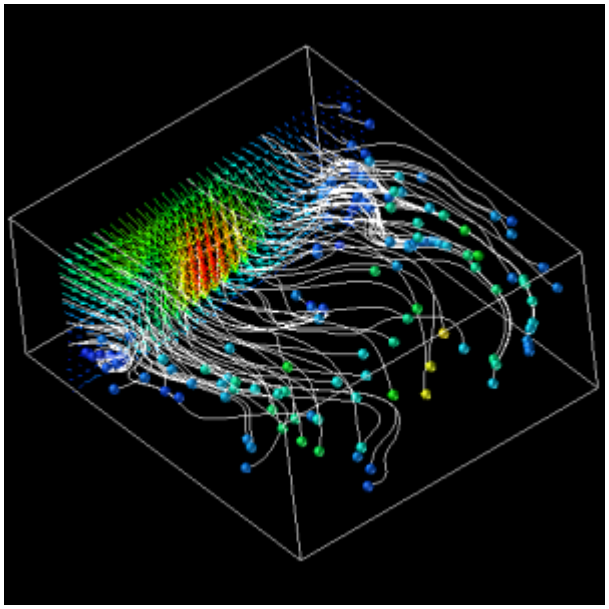
GD kit threading

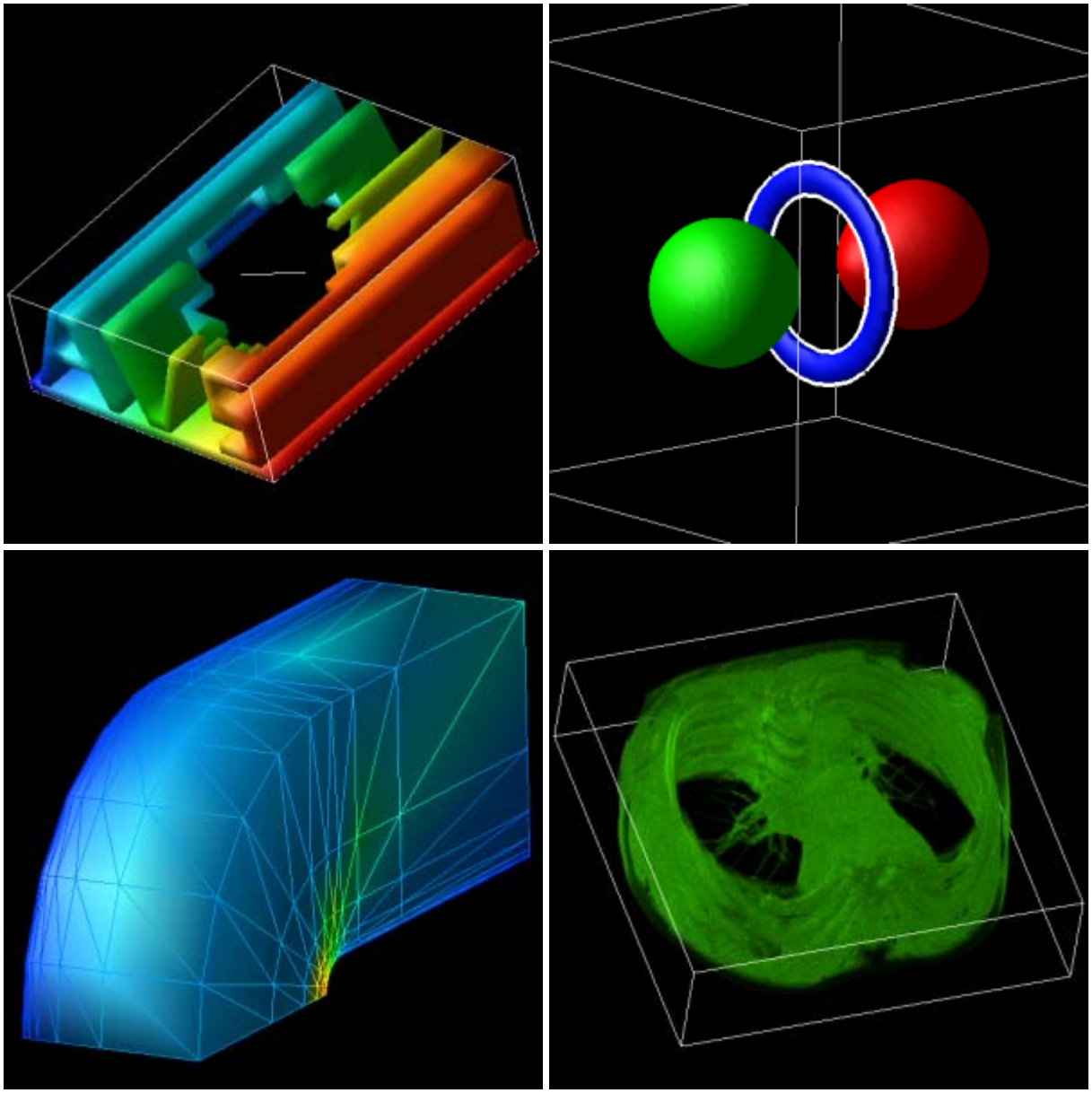
The speed of the GD kit on multi-core and multi-CPU systems has been increased through use of threading. This functionality is enabled by default, but can be disabled by setting the XP_GD_NO_THREADS environment variable or configuration file setting. The number of threads is by default the number of cores on the system minus one, unless overridden by the XP_GD_NUM_THREADS setting.

New macros

Please see the "What's New" library for example applications demonstrating these new macros.







- **Arbitplane** - plane geometry which can be defined by various combinations of points, vectors & angles
- **HiAxis3D** - 3D axis geometry with enhanced label formatting options
- **Multi_Files** - select a directory, then a sequence of files with an extension, reorder & loop through the files
- **MultiViewer3D** - 3D viewer with two scenes side-by-side, transformed together and saved as a single image
- **Read_DICOM** - read DICOM format medical images
- **Read_Tiff16** - read a TIF file containing 16-bit data
- **Read_Tiff16_Vol** - read a sequence of TIF files containing 16-bit data and construct a volume field
- **Uviewer3DAxis** - 3D viewer with an integrated XYZ axis
- **Uviewer3DUnitScale** - 3D viewer with an integrated unit scale
- **Write_Div_Image** - output a tiled array of images from a view
- **Write_KML** - write a field in KML format (Google Earth)
- **Write_STL** - write a field in STL format, ASCII or binary

- **Write_Text** - write a field in plain text format
- **Write_UCD** - write a UCD file, support binary and time-dependent formats
- **calc_normals** - calculates the normal vector of 2D cells and places the result in a cell data component
- **crop_area_box** - an interactive method to crop an unstructured field using a drag rectangle
- **crop_cylinder** - crop inside or outside a cylindrical region of a field
- **crop_orthobox_2way** - enhanced version of crop_orthobox with finer control
- **crop_orthoslice** - crop a thin slice of a field
- **crop_sphere** - crop inside or outside a spherical region of a field
- **cut_arbitplane** - cut macro using an Arbitplane geometry
- **data_to_rgb** - convert node data values directly to RGB color
- **isosurface_segment** - separates the output field from an isosurface according to the connection information - each segment can be highlighted
- **make_node_data** - add up to 7 additional node data components representing various aspects of a field
- **merge_fields** - an enhanced version of combine_sets_ARR which supports multiple components and cell types
- **quad_to_tri** - convert quad, quad2 and tri2 cells to tri
- **point_iso_struct** - draw an isosurface as illuminated points instead of trianguler cells
- **point_probe** - display nodes as a point mesh, then pick a point to display its value and coordinates
- **point_sprite** - render a point mesh of high-quality spheres (requires OpenGL 2.0 capable hardware)
- **slice_arbitplane** - slice macro using an Arbitplane geometry
- **surf_plot2** - an enhanced surface plot which can use the X or Y direction and null data
- **surface_point** - draw a field as illuminated points
- **threshold_all** - threshold across all node data components
- **threshold_cell_all** - threshold across all cell data components
- **time_advector** - create a particle trace for time-dependent data
- **ucd_area_ctrl** - use crop_area_box to crop a field, select each cropped region & crop another field using the same region

Updated macros

- **isoline** - the iso levels can now be entered manually
- **streamlines** - add a normalize parameter to inhibit the scaling of ribbon widths due to the divergence
- **tube** - can now use cell data for coloring and scaling