



## 2.1.0

- BUG 267 Fixed a crash that might occur for transparent isosurfaces and isovolumes if they contain lines and non-simple transparency is used.
- BUG 265 ElementHighlighter: Element edges was drawn with undisplaced nodes. Use displaced coordinates if part is shown with a displacement result.
- BUG 264 Render to image (with custom size) on a new view with graded background did not work. This regression was introduced by BUG243.
- BUG 261 Fixed crash than could occur when loading data from reader failed during UnstructGridModel::updateVisualization()
- BUG 260 Fixed bug where cee::vis::Font::createTrueTypeFont(..) didn't return NULL if unable to load the given file. Note that this function is deprecated and will be removed in the future. Use cee::vis::TrueTypeFont(const Str&, unsigned int) instead.

## 2.0.0

- BUG 258 Markup parts might not be removed from the scene if there are transparent parts and not simple transparency.
- BUG 257 Sphere and comet particle traces did not use the current clipping settings.
- BUG 253 Labels with DRAW\_ONLY\_VISIBLE or DRAW\_WITHOUT\_ZBUFFER\_TEST draw mode caused an undesired overdraw effect when lots of labels were positioned on top of each other.
- BUG 249 Fix asserts in SurfacePathQuery
- BUG 248 Isovolumes did not work when computed/mapped scalar had per element node results and no nodal results was available.
- BUG 247 ElementsQuery::elementSurfaceVertices() caused an assert for beams and point elements.
- BUG 246 MarkupPartLabels did not respect the clipping planes (View.clipping and ug cutting planes with clipping).
- BUG 243 Optimized handling of clipping. Markup model only updates display model when the clipping has been turned on/off. View defers updates of models to next redraw to avoid update of all models when updating clipping planes by removing all and adding them again.

## 1.4.9

- BUG 229 Fixed misidentification of third principal vector derived result in DataSourceReader. It was mistaken with third principal value and was not being computed
- BUG 226 Vectors on cutting planes did not respect the color specified in the VectorSettings when in SingleColor mode.

## 1.4.8

- BUG 223 Rubber band zoom (Camera::rubberbandZoom()) caused an assert if the field of view was too large.
- BUG 220 Scalar fringes produced flickering on Quadro graphics cards on models with constant values on element faces when having uneven spaced legend levels, and the constant value was right on the value for a level change.

## 1.4.7

- BUG 228 Fixed bug in View::renderToImage(unsigned int width, unsigned int height, Image\* image). Due to texture format requirements, the width needs to be a multiple of 4. The given width is now corrected on the spot to comply.
- BUG 215 Scalars mapped on particle traces were not updated when when scalar settings were changed and the SCALAR\_SETTINGS flag was used in updateVisualization().
- BUG 214 Updating an isovolume with the ISOVOLUME flag (updateVisualization(cee::ug::UnstructGridModel::ISOVOLUME)) caused an assert.
- BUG 212 Setting above and below colors did not work for zero range legends (min == max).
- BUG 211 No result legend tickmark override (cee::ug::ModelSettings::setColorLegendNoResultOverrideTickMarksText()) did not reset the legend properly when set back to default ("").
- BUG 210 The continuous color legend was rendered in a single color when having a zero range (min == max).
- BUG 197 ScalarSettings::setLegendVisibilityMode(ScalarSettings::ALWAYS) did not work as intended.



BUG 196 Proper handling of scalars/color legends with zero range.

The single value is shown on all tickmarks, and the model gets the color from the first level. Fixes regression with color legends with all level values set to the same value getting all red color.

## 1.4.6

BUG 195 OverlayTextBox alignment did not work correctly when alignment was changed on existing text box

BUG 194 VTFx export using DISPLAY\_MODEL\_ONLY model exported wrong node positions when having scaled displacements.

BUG 187 VisualizationPartQuery::visibleNodes() and ::visibleElements() now uses size\_t for indices.

BUG 184 Fixed crash when highlighting isovolumes

BUG 183 Color legends with logarithmic mapping and custom level values for the Color Mapper type FILLED\_CONTOURS did not work as expected when properties were loaded from VTFx files.

BUG 181 ScalarSettings::autoRangeModel set to VISIBLE\_ITEMS required two calls to updateVisualization() and crashed if an adaptive model was used. Both issues have been fixed.

## 1.4.5

BUG 178 Texture mapping in the geometry model was sometimes wrong (mixup of textures) after adding, removing and re-adding parts with texture effects.

BUG 175 Fixed several issue with logarithmic mapping in the UnstructGrid model. Now works both for all 3 color mappers and also for export/import to VTFx, as well as support for legacy VTFx files.

BUG 167 Halo highlighting of cutting planes produced a lot of highlighted interior edges when clipping was enabled.

## 1.4.4

BUG 159 Reference counting in C# did not work for factory methods (methods producing objects but not keeping an internal reference) and some other cases where references were returned from C3DC.

BUG 158 Changing node positions and then using the updateVisualization(cee::ug::UnstructGridModel::NODE\_POSITIONS) did not update the bounding boxes of the UnstructGridModel.

BUG 153 Flat shading failed on hardware without GL\_EXT\_gpu\_shader4 extension (e.g. on VM's using Mesa based rendering).

BUG 151 Mode shape animation setup failed if setup twice with the exact same parameters.

BUG 142 regionIntersect() and polygonIntersect() did not respect the part visibility in the UnstructGridModel.

BUG 128 ElementHighlighter only showed the initial step when using mode shape animations. The ElementsQuery class did not support mode shape animations.

Both have been resolved and we added ElementsQuery(const UnstructGridModel\* model, size\_t frameIndex, size\_t geometryIndex, int partId) constructor to use when having mode shape animations.

## 1.4.3

BUG 150 Display model is updated when nodes visibility is changed after setting the model's grid

BUG 144 DataElementSetGenerator did not honor the part visibility when creating sets (did not consider PartSettings::setVisible() flag).

BUG 143 DataPoints parts in the geometry model did not respect the transformation matrix of the part and the model.

## 1.4.2

BUG 125 The color legend was hidden if a scalar result was set as fringes in the modelSpec but no parts had fringesVisible set to true, but it was still used for scalar filtering.

This behavior has been changed and the color legend is visible for any scalar result that is used for scalar filtering.



BUG 124 Scalar filtering (ScalarSettings::enableFringesElementFiltering()) did not work in combination with mode shape animations.

BUG 123 OverlayItems with absolute position had a 3 pixel offset to the specified position causing them to be slightly off the given position.

BUG 122 ImportCae: Got rid of a useless assert that was triggered when displaying results on the top layer of a mixed element model. The assert occurred because volume elements are not compatible with layers - for these, results are stored in the bottom layer and other layers are undefined.

## 1.4.1

BUG 120 ImportCae: Fixed problem with reading results from ODB file where system state is local deformed. (STATE\_ROTANG)

BUG 118 Fixed problem with setting fonts as properties in C#

BUG 117 Fixed element highlighting issue in PyQt demo app when using rubberband picking

BUG 116 Enabled LogDestination to be inherited i C# to be able to create your own log destination.

BUG 109 Plot2D: Fixed bug with axis range /auto range not being set correctly.

BUG 107 Fixed wrong value returned by OverlayPlot::yValuesMaximum().

BUG 91 Fixed bug in picking. When zooming out, returned picking information was flawed in some situations. The bug was to consider that ray intersection points with lines of the model were on the hit lines, not the ray. Due to this, the determination of the picked items was producing unexpected results.

## 1.4.0

BUG 89 updateVisualization(NODE\_POSITION) and mode shape animation setup with partial generation crashed when using setUseShaderComputedFlatNormals() in UnstructGridModel.

BUG 87 cee::ug::setOptimizePartRendering(true) and animations with DataResultTransformation results did not update the parts with eyeLift != 0.

BUG 86 DataSourceMemory::updateDirectoryFromStates() did not create displacements info for the missing displacement results.

## 1.3.1

BUG 84 MarkupModel::rayIntersect could cause an assert if an item was added and no rendering of the items was done before doing a rayIntersect().

BUG 83 GeometryModel::rayIntersect() did not handle clipping planes if not created from Camera. If a Ray was manually created and the min and max distances in the ray not properly set, the rayIntersect() method in GeometryModel would not honor clipping planes. This has now been fixed.

Also added two new methods for convenience: UnstructGridModel::rayIntersect(int x, int y, const vis::View& view, HitItem\* hitItem) and GeometryModel::rayIntersect(int x, int y, const vis::View& view, HitItem\* hitItem). Both methods honor clipping planes.

BUG 82 Transparent parts (both with opacity and transparent textures) did not respect the lighting setting when using DepthPeeling or WeightedAverage transparency modes, and was thus always rendered with lighting on.

BUG 81 Image situations inserted into VTFx tagged plugins has the wrong aspect ratio

## 1.3.0

BUG 76 Warning (log) and not an assert if the geometry index in the VTFx file properties are illegal. Added DataSource::geometryCountPerState()

## 1.2.1

BUG 75 Fixed Qt "pro" files for use with Qt5. Added missing opengl libraries to link line.

BUG 74 Invisible pre-computed isosurfaces, isovolumes and particle traces exported to VTFx and thus visible in the 3D Viewer.

BUG 73 Fixed bug in DataSourceReader::reload(), ensuring that tensor-derived results are present when leaving the method

BUG 72 Invisible pre-computed cutting planes exported to VTFx and thus visible in the 3D Viewer.



BUG 70 Fixed problem with creating a custom DataReader in .NET

BUG 69 Assert when having iso surfaces (and not iso volumes) and a custom `cee::ug::DataReader`

## 1.2.0

BUG 68 `UnstructGridModel::rayIntersect()` returned an intersection point normal only if the picked item was of type `cee::ug::HitItem::PART`. No information was provide for `cee::ug::HitItem::CUTTING_PLANE`, `cee::ug::HitItem::ISO_SURFACE` or `cee::ug::HitItem::ISO_VOLUME`

The method now ensures that a normal is provided for types of items. The information is conveyed to `UnstructGridModel::rayIntersect()` through an extension of `VTSelectItem`, which now includes a new `intersectionPointNormal` public member. For parts, this normal is set in `DoPickCvfParts`, in the calls to `VTPart::DoPickSourceTriangles()` (calling `cvf::Ray::triangleIntersect()`) or `VTPart::PickDrawableGeo()` (calling `cvf::DrawableGeo::rayIntersect()`). Note that the normal is set to zero when picking lines or points in `DoPickSourceLines()` and `DoPickSourcePoints()`. For cutting planes, isosurface or isovolumes the normal is set when parsing for picking through the `VTFeatureExtractionResults` of the object, in `VTFeatureExtractionResult::Pick()` (also calling `cvf::DrawableGeo::rayIntersect()`). All methods that collect the intersection point normal ultimately rely on `cvf::Ray::triangleIntersect()`.

BUG 67 Fixed bug in `sampleScalar()` and `sampleVector()` methods on `CuttingPlane`, `Isosurface` and `Isovolume` which reported wrong results.

BUG 66 Fixed uninvertible projection matrix issue by avoiding field of view angle = 180.

BUG 64 Assert when computing min/max of a scalar which is not allocated on all parts

BUG 58 Scalar results with a very narrow range were not yielding the expected number of fringes.

BUG 57 `MarkupPartText3d` items did not render properly (not respecting z buffer) when some parts were transparent and non-simple transparency was used. Also general problems with transparent parts in the Markup Model.

BUG 55 Fixed appending compatible cases to `Vtfx` files

BUG 54 Fixed crash observed when querying values from a data source in `DataSourceQuery`. The method now handles points on which result values are undefined

BUG 53 Fixed bad assert (in debug) in `setAmbientIntensity/setSpecularIntensity` in `cee::geo::EffectColor/EffectTexture/EffectFrontAndBackColor`.

BUG 52 Qt demo app crashed when selected scalar is cleared after one has been selected

BUG 49 `cee::ug::ModelSpec::clear()` did not clear `statelds`, `vectorlds` and `setlds`

BUG 45 Removed obsolete error message shown in some cases when updating the Geometry Module model

BUG 44 Fixed bug in the Geometry Module causing a removed Part not to be removed when adding a new Part before updating the model, in some cases.

## 1.1.1

BUG 48 Derived results missing after `DataSourceReader::reload()`

BUG 46 Removed unnecessary assert on number of polygons containing a path point  $\leq 1$ . The case of multiple polygons being found on a path point is handled.

BUG 41 Bounding box query before rendering a `MarkupModel` (i.e creating the underlying display model) did not work.

BUG 40 `SurfacePathQuery` fixed for queries for scalar values on `DataBases` with no `DataInterface`. Queries on vectors or tensors are unsupported in this version

BUG 39 `UnstructGridModel::clearVisualization()` asserted when called on a model without any frames.

BUG 37 Changing visual attributes (e.g. surface color) on cutting planes, isosurfaces and isovolumes and then calling e.g. `updateVisualization(CUTTING_PLANE)` did not update the cut/iso/isovol.

BUG 36 The Mfc tutorial runner launched the texture tutorial when selecting the "custom reader" menu entry.

BUG 31 `UnstructGridModel::updateVisualization()` with `NODE_POSITIONS` flag did not work with `optimizePartRendering` was set to true.



---

## 1.1.0

- BUG 34 Transparent triangle parts in the Markup Model were opaque when rendering with shaders (hardware based OpenGL).
- BUG 32 In some setups the topmost and the second topmost label on the color legend in the UnstructGridModel would overlap. This happened when some of the tick mark labels were omitted due to restricted height of the color legend.
- BUG 30 Added workaround for bug in Intel HD Graphics driver related to FrameBufferObjects that would give INVALID\_OPERATION and distort the view when having multiple viewers and e.g. using depth peeing transparency.

---

## 1.0.3

- BUG 29 Transparency mode "DepthPeelingFront" did not work as expected on some Amd FirePro cards.
- BUG 28 Dual Depth Peeling and Transparent Weighted Average failed on Intel HD 4400 with shader compile error.
- BUG 27 Cutting plane clipping is not updated when using UnstructGridModel::updateVisualization(CUTTING\_PLANE)

---

## 1.0.2

- BUG 22 Automatic resizing of color legends did not work as expected when one UnstructGridModel was shown in multiple views with different height.
- BUG 21 Overriding effects in the Geometry Model sometimes caused all parts with the same original effect to be affected, not just the part(s) that had extra (override) effects.
- BUG 20 Depth Peeling not working on some legacy graphics cards with OpenGL version 3.0