

Release Notes

This document lists the most relevant changes, additions, and fixes provided by the latest MeVisLab releases.

Version 3.1.1 Stable Release (2018-12)

New Features

- C++: Updated the `Eigen` library to version 3.3.5. This was necessary because newer updates of Visual Studio 2017 generate a compile error when including certain header files of this library.
- C++: Support CUDA 10 in project generation.
- C++: Added new template class `TypedBaseField` to ML library, which allows for convenient typed access of the contained Base object.
- C++: Added method `clearTriggers` to the `SoPointingAction` class.
- C++: Make libraries `MLGraphUtilities` and `SoCoordUtils1` usable from own projects.
- C++: Added macro `SO_NODE_ADD_ENUM_FIELD` as a shortcut for `SO_NODE_SET_SF_ENUM_TYPE` and `SO_NODE_ADD_FIELD` to `Inventor`.
- The `Field` control got a `browsingGroup` attribute, which allows to store last used directories for different purposes separately. There is also a new optional parameter of the same name for the methods of the `MLABFileDialog` scripting class.
- Added a scripting wrapper for some of the `OpenVDB` functionality. Using the wrapper can be more efficient than using the `OpenVDB` modules, since several level-set operations can be chained. See class `MLOpenVDBToolsWrapper` in the Scripting Reference for details.
- Added function `addQuadPatchFromNumPyArray` to WEM scripting wrapper, and fixed `addTrianglePatchFromNumPyArray`, which didn't associate the nodes to the faces.
- Also added function `addPatchCopy` to WEM scripting wrapper to copy a patch from another WEM.
- Module `DicomReceiver` got a new field `additionalStoreSCPOptions` to specify additional command line options.
- The minimum and maximum values of the `BoundingBox` module can be activated separately with checkboxes now.
- Added `isMouseOver` field to module `SoMenuItem`.
- Added `relativeAnchorPoint` field to module `SoFixedMenu` (used when not in pop-up mode to position menu items at a relative position of the viewing area).
- Added `performPreHitTest` field to module `So3DMarkerEditor`.
- The `rotation` field of module `WEMModify` is a real `MLABRotationField` now instead of just a `MLABVector4Field`, which means that you now can get/put an `Inventor SbRotation` out of/into it in Python scripting.
- Improvements for module `CreateRTStruct`:
 - Added field `treatCollinearAsPlanar` which allows to change how collinear CSOs are handled.
 - Input image modality is not restricted to CT or MR any more.
 - An error is reported if the input `CSOList` contains ungrouped CSOs.
 - Enhanced printed information about problems with referencing slices.
 - Make sure that the `NumberOfContourPoints` is consistent with the number of values in `ContourData` when writing CSOs.
- Windows: The Mesa OpenGL fallback is always added to stand-alone applications if you don't set `WITH_MESA` to 0 in your `.mlinstall` file.

Fixes

- C++: Fixed new ML list field method `updateValueAt` not updating the value.
- C++: Fixed that some memory allocation errors might fail silently.
- C++: Backported a Qt patch for `QProcess::start`: Avoid a name collision in named pipes for processes started in parallel.
- `ComputedAttributes` in an `ItemModelView` control didn't update when one of the attributes in the expression changed.
- The `ColorEdit` control does not set the color to black anymore if the color dialog is cancelled.
- Removed the zoom restriction from the `Resample3D` module (factors above 1000 were set to 1).
- Modules based on `OpenVDB` crashed if used on WEMs with more than one patch.
- Fix handling of keyboard shortcut events in `RemotePanelRendering`.
- Fixed that `SoDiagram2D` used the color of the first style to draw the last curve when `stylePaletteMode` was `STYLE_PALETTE_NAME`.
- Fix crash under certain circumstances when removing the `SoView2DCSOEditor` module from a network.
- Fixed that the output of module `SoPostEffectBackground` in `OffscreenRenderer` turned all black when opening an internal network.
- Fixed menu item placement in `SoFixedMenu` and `SoBorderMenu`: Items attached to lower/left border jumped around when resizing.
- Improved blending of `SoView2D` font rendering with background.
- Fixed an infinite `Inventor` loop in module `View2DTouch`. Also fixed usage of unknown field `hiResRendering`.
- Ported some Python tutorial modules to Python3: `Tutorial_BasicPython`, `Tutorial_MeVisLabAndPython`, `Tutorial_AdvancedPython`.
- Ported module `CLIImporter` to Python 3.
- The `ChangeSet` Python class would cause an exception when the MeVisLab module on which the class was applied was already deleted when the destructor was called.
- Fixed some bad handling of OpenGL clip planes (which probably nobody really noticed).
- Number inputs in module GUIs didn't allow to type in (temporarily invalid) incomplete floating point numbers.
- MATE: Do not place cursor on previous line when de-indenting.
- MATE: Python refactoring with rope updated to new version 0.10.7 to fix that extracting a function failed.
- MATE: Disabled pylint checks were not persistent.
- MATE: Fixed that the positions of results of the **Find in Files** dialog were wrong if a result line was shown truncated.
- Tools like MATE don't write to the same log file as MeVisLab anymore.
- macOS: QuickLook and Spotlight plugins work again.
- macOS: Hide "Show Tab Bar" menu items added for 10.12 and up.
- macOS: Movie generation improvements (vertical flip of movies generated from image files, presentation time calculation for movies generated from memory images).
- macOS: Fix crash in IDE if size of recent files list is set to zero on system..
- Linux: Improved CUDA detection.
- Linux: Translation tool could not be opened directly from a multi-language module.
- Documentation fixes.