



パーソナルプリポストプロセッサ

GiD

Version 16

リリースノート

一般

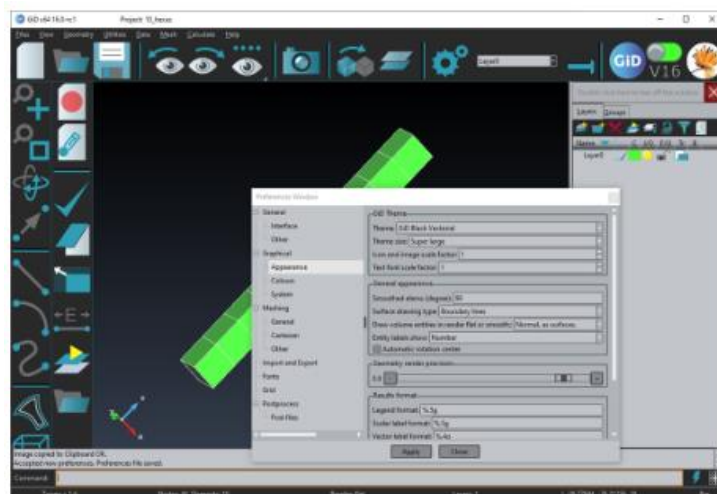
- ✓ ログインユーザーはモデルのストレージとしてクラウドを利用することができます。



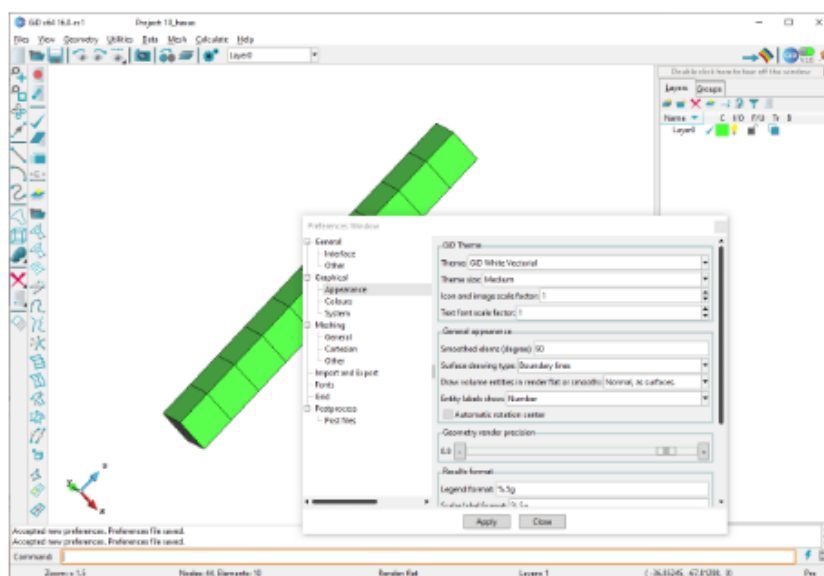
- ✓ “Utilities – Status”から、共有フォルダ（テンポラリ、User Settings など）を開くためのボタンにアクセスできます。
- ✓ 計算の前に名前を付けて保存する必要がなくなりました。保存していないモデルのためにテンポラリフォルダが使用されます。
- ✓ ファイルブラウザ上での圧縮/解凍がサポートされました。
- ✓ Html ヘルプのファイルサイズが以前のバージョンから改善されました。
- ✓ ログインサーバにリクエストからの応答問題が発生した場合、GiD のロックが回避されるようになりました。
- ✓ Nastran, Abaqus, Ls-dyna のプロブレムタイプ使用時にパスワードが不要になりました。



- ✓ GiD のリスタート時に、テーマが変更されなくなりました。
- ✓ ブラックテーマが改善され、高解像度画面で美しく描画されるようになりました。



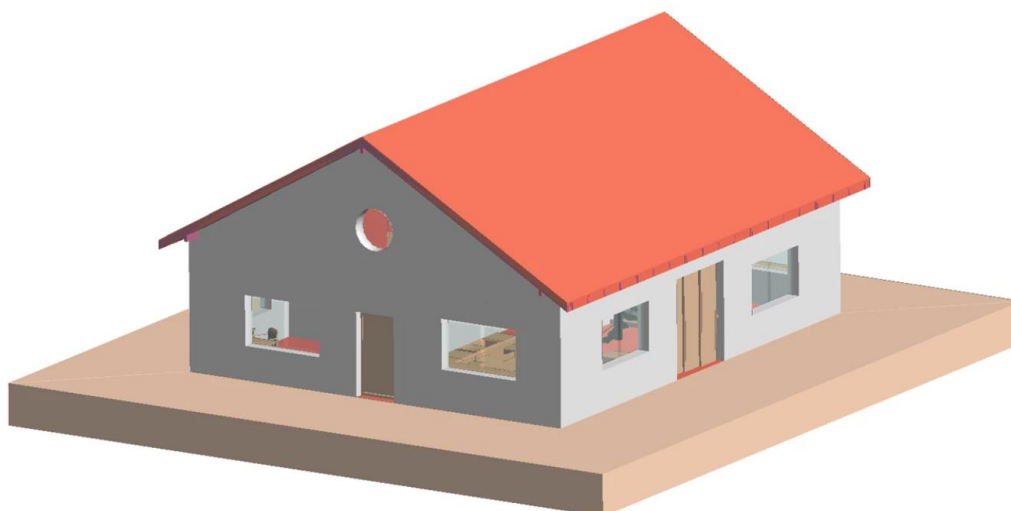
- ✓ 「GiD White Vectorial」テーマが新しく追加されました。



- ✓ MacOS 10.15 Catalina に対する USB ライセンスへのアクセス問題が解決されました。
- ✓ MacOS retina がサポートされました。

プリプロセス

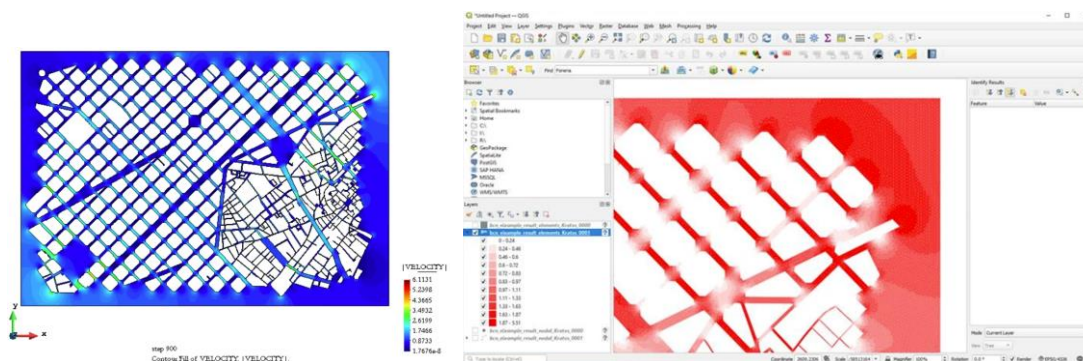
- ✓ 各ジオメトリエンティティに作成されているメッシュを保持することができるようになりました。様々なオプションを利用することができます。（メッシュが既に作成されている場合、リメッシュを避ける、メッシュをアップデートする、ジオメトリとともに、メッシュもコピーする、など）
- ✓ IFC ジオメトリフォーマットのインポートをサポートしました。



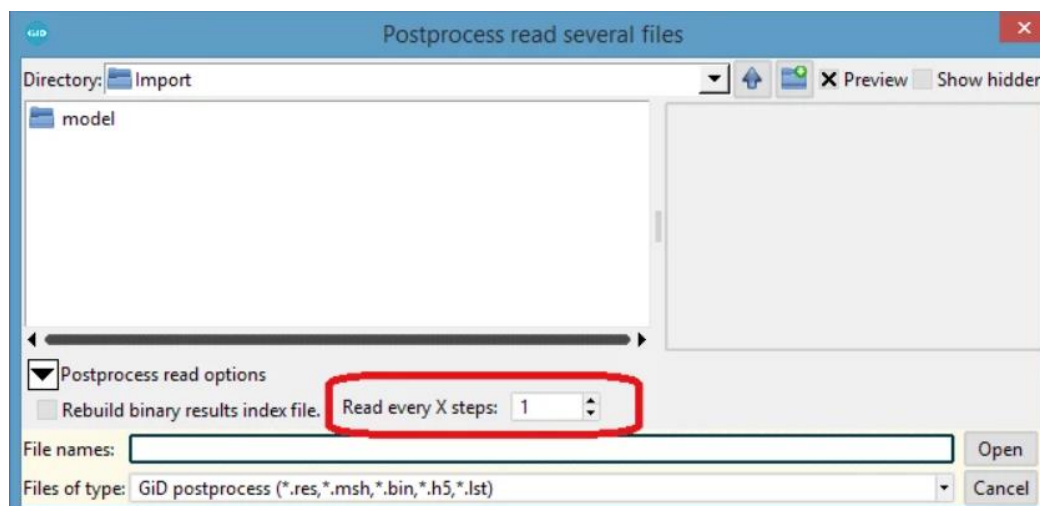
- ✓ Undo 機能が強化されました。(Redo 機能の追加、ひとつ前のコマンドの Undo、バックアップを開始点として使用、モデルセーブ時も履歴が失われない、Undo の複数ウィンドウからのコマンドのコピー、など)
- ✓ 接触ボリュームの完全性がチェックされます。また、シンプルなケースでは自動的に修復されます。

ポストプロセス

- ✓ 結果を shape file フォーマット (GIS) で出力されるプラグインが搭載されました。



- ✓ iso-surfaces を Cuts に変換するオプションが追加されました。
- ✓ 結果キャッシュのメモリ不足に関する不具合を修正しました。
- ✓ ボリュームレンダリングに関する機能が強化されました。
- ✓ 複数のメッシュをカットする際の不具合を修正しました。
- ✓ 複数の VTK ファイルをインポートする際の問題を修正しました。
- ✓ HDF5 で結果の読み込み/書き出しする際に発生するエラーを修正しました。
- ✓ 結果読み込み時にステップをスキップすることができるようになりました。



カスタマイズ

- ✓ Tcl/TK のバージョンが 8.6.12 にアップデートされました。
- ✓ <GiD>scripts/tk/demos/DEMO_widgets.vbs は tk ウィジェットデモのコレクションとそのソースコードを表示します。(Windows)
- ✓ GiD/tk_widgets_demo スクリプトが追加され、tk ウィジェットデモのコレクションとそのソースコードが表示されるようになりました。(Linux, macOS)
- ✓ カスタムライブラリ
 - ◇ Customlib プロブレムタイプは xml spd ファイルに埋め込まれる必要がなくなりました。
 - ◇ correct_local_axes_with_lines コマンドで、いくつかのカーブ上の方向に基づいて割り当てられたローカル軸を修正することができます。
- ✓ GiD-Tcl イベント
 - ◇ GiD_Event_BeforeCalculate, GiD_Event_AfterReadPreferences, GiD_Event_AfterSavePreferences, GiD_Event_BeforeExit, GiD_Event_AfterDataManagerSetCloudFolder
- ✓ Tcl コマンド
 - ◇ GiD_AccessValueAssignedCondition ?-field_index? set|get <condition> <over> {<question>|<field_index> ?<value>? ... <question>|<field_index> ?<value>?} <entities>
 - ◇ GiD_GetUserSettingsCommonDirectory ?-create_folders?, return the main user GiD folder
 - ◇ GiD_Project backup save|read ?-ignore_must_save_flags? <name>
 - ◇ GiD_Mesh list face (e.g. "GiD_Mesh list -normals_dotprod_threshold -0.9 face" to identify problems)
- ✓ Tcl サブコマンド
 - ◇ GiD_Info bounding_box new optional flags ?-geometry|-mesh|-post? ?-layers_off_also?
 - ◇ GiD_Info mesh nodes | elements new option -array2 :plane list の代わりに、objarrays の list として結果を返します。
 - ◇ GiD_Info mesh -post -step_index \$idx_step -set_name \$name Elements \$element (new optional flag -set_name \$set_name)
 - ◇ GiD_Info mesh -post ?-step_index <idx_step>? ?-step_value <step_value>? (new optional flag '-step_value')
 - ◇ GiD_Info variables ?-mesh? ?-expand_array_names?
 - ◇ GiD_Info check surface <entity_id> selfintersection ?-tolerance <tolerance>? (new -tolerance optional flag)

- ◇ GiD_Info check mesh contact_elements_connectivities : 接触要素の完全性をチェックします。
 - ◇ GiD_Info postprocess get cur_step_index : カレントステップの整数 index を取得します。
 - ◇ GiD_Groups get all_properties <group> and GiD_Layers get all_properties <layers>
 - ◇ GiD_Geometry get: 新しいサブオプション label_on | selected が追加されました。
 - ◇ GiD_Geometry edit point|line|surface|volume <num> material|label_on|selected <value>
 - ◇ GiD_Mesh get: 新しいサブオプション label_on | selected が追加されました。
 - ◇ GiD_Mesh edit node|element <num> material|label_on|selected <value>
 - ◇ GiD_Project transform_problemtype <new_problemtype>
 - ◇ GiD_Sets get command: 値を取得するための element_type と element_num_nnodes オプションが追加されました。
 - ◇ GiD_Result get new optional flag -ignore_no_result : 設定されていない結果値を無視することができます。
- ✓ パッケージ
- ◇ objarray パッケージの -binary <data> オプションが追加されました。
 - ◇ objarray foreach <var_name> <objarray> <body> で、配列の全ての値を簡単にループさせることができます。
 - ◇ Tcl パッケージ vfs が追加されました。
 - ◇ promise パッケージが追加され、promises と async/await 構文で非同期プロセスを実行できるようになりました。
 - ◇ tksvg パッケージが追加され、svg フォーマット(scalable vector graphics)のベクトル画像をサポートするようになりました。
- ✓ 新しい GiD の変数 ShowFigures(BoundingBoxLines), ShowFigures(BoundingBoxSurfaces), ShowFigures(BoundingBoxVolumes) を利用して、エンティティのバウンディングボックスを描画することができます。
- ✓ GidUtils::OpenWindow , GidUtils::CloseWindow , などで、従来のデータウィンドウを処理することができます。