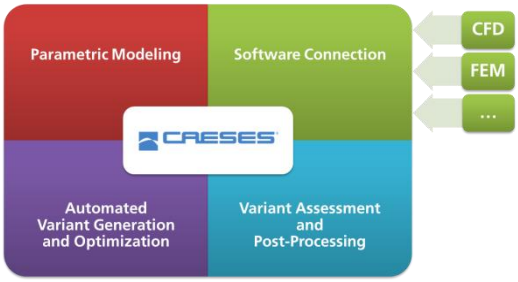
External Software

CAESES allows coupling and integration to external simulation software that is used in your design process. In particular, tools for flow simulation (CFD) or structural analysis (FEM) can be plugged-in and controlled from within CAESES.



This overview is generalized and the steps shown can be applied to any commercial and non-commercial external software. Basically, the software should be merely able to run in batch mode.

Geometry from CAESES is exported to the software and template input files that control the software can additionally be manipulated for each design. Finally, result data such as simulation results are then directly loaded into CAESES. They can be accessed and utilized by the user, for instance, in optimization loops – being the natural next step when the simulation is connected.

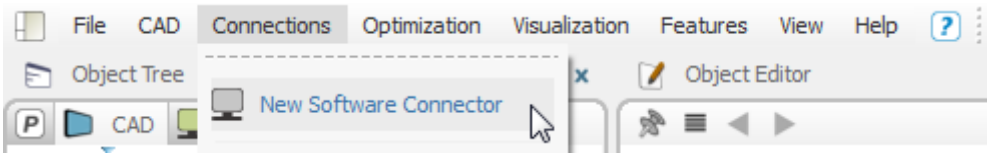
The basic setup to couple with external software is as follows:

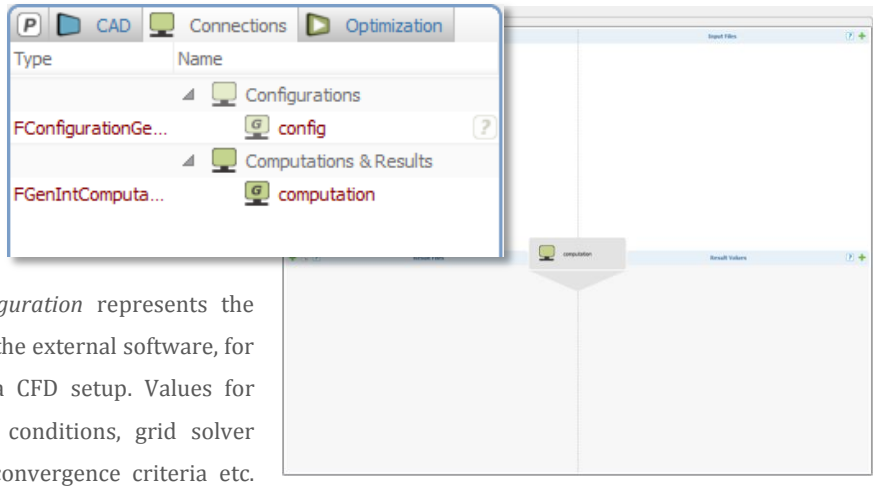
- Simulation software that runs in batch mode (script, executable)
- Optional ASCII input files that control the external software
- ASCII output files, in particular containing key result values such as pressure loss, relevant coefficients etc. (*csv*, *txt*, etc.)
- Optional output screenshots that will be shown in the picture viewer of CAESES.
- Optional output files (*vtk*, *tecplot, ensight, OpenFOAM*) for interactive 3D post-processing such as plane cuts, streamlines, etc.

1 Software Connector

The *software connector* is the administration object that configures the data exchange between CAESES and the external software.

- From the menu, select *connections* > *new software connector.* This creates a new *configuration* and a new *computation* object (see the *connections* tab).





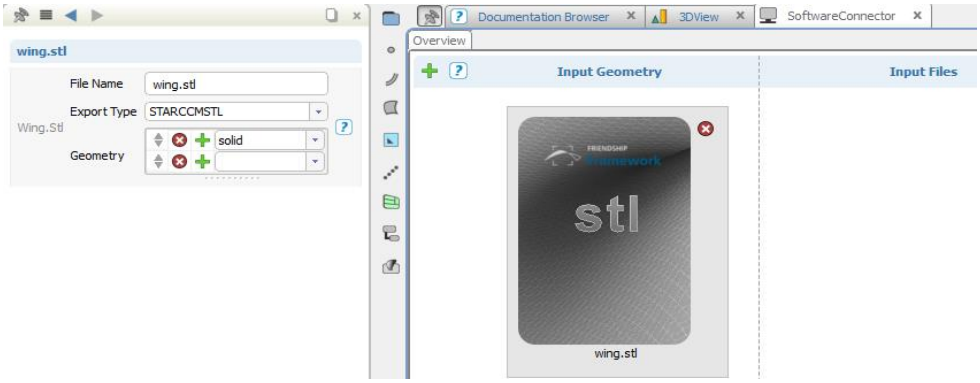
The *configuration* represents the setup for the external software, for instance a CFD setup. Values for boundary conditions, grid solver settings, convergence criteria etc. can be set in the configuration. In addition, any information about input and output files will be stored in the configuration (this is automatically done via the software connector). The explanation for how to add entries to a configuration is given in a later step of this tutorial.

The *computation* will contain information about the external application (e.g. executable path, batch mode arguments, constraints, SSH settings for distributed computing etc., see step 9).

2 Input Geometry

The software connector window contains a field that is called *Input Geometry*. This is where you configure the geometry export from CAESES to the external software.

- Click on the *CAD* tab of the tree to access the geometry.
- Select your geometry objects in the tree and drag & drop it from the tree into the first quadrant *Input Geometry* (see screenshot).
- In the object editor, set a file export name and choose a file format.



✓ In the screenshot example, a colored STL format is utilized which allows the automatic detection of regions such as inlet, outlet and walls with different colors, within the CFD software. The color assignments have been done in a previous tutorial. Simply drag & drop the trimesh or solid object into the software connector.

✓ Generally, all input data of a connection gets exported/copied into the current design directory. If you are in the baseline (i.e. this is not a design engine run), the path is as follows:

<project directory>/<project name>/manual_results/<computation name>.

For variations and optimizations, this path becomes:

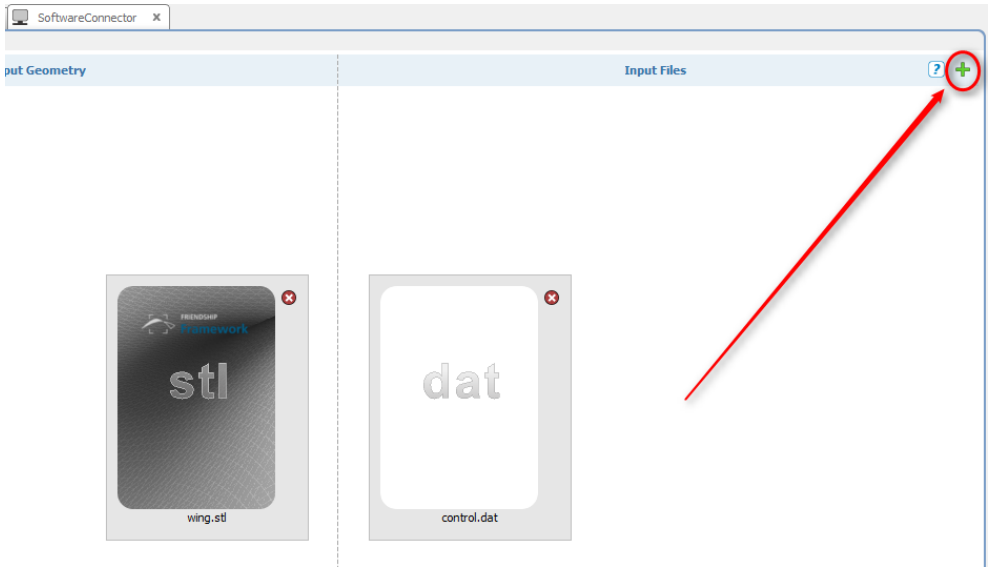
<project directory>/<project name>/<design engine run>/<design name>/<computation name>.

Typically, external software will be triggered from within the working directory. This automatically corresponds to the design's computation directory of CAESES so that all required files will be usually found by the external software.

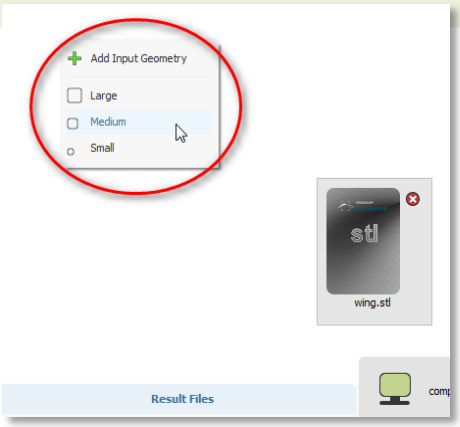
3 Input Files

Let's assume that for a CFD run we need a set of input files to be automatically copied into the current design directory (i.e. baseline/variant directory) where the batch mode will be triggered:

- Add input files to the connector via the "+" icon at the top right corner. All files in this second quadrant will be copied later on.

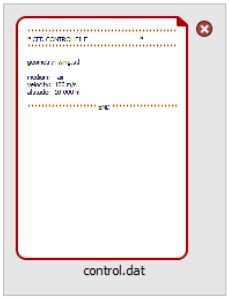


✓ There are 3 different icon sizes available via the context menu of the software connector.



4 Template Input Files: Replace Fixed Numbers

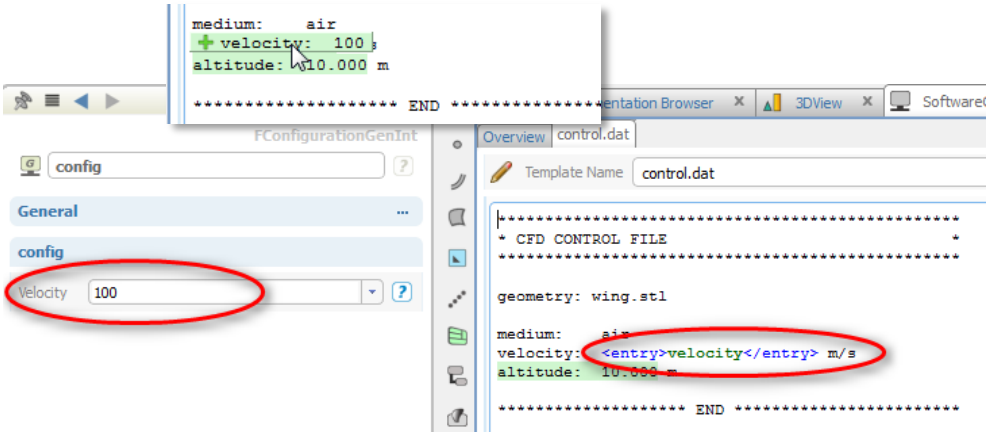
If you need to modify your input files before the CFD run is started (e.g. replace fixed values in the ASCII file with variable ones), convert it to a template file which is then a copy of the original file:



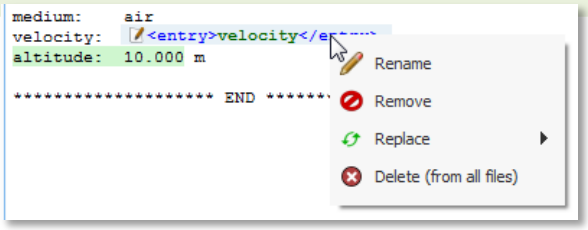
- Double-click on the file icon (here: control.dat) so that a template editor is opened. The icon of the input file also changes slightly.

If you want to replace a number by a variable value:

- Click on the green-colored number detection field that highlights available numbers in this file (this creates a so-called *configuration entry*).



✓ Manipulating the auto-detected entry: Click on the detected entry again in order to get the context menu for more options (rename, remove etc).



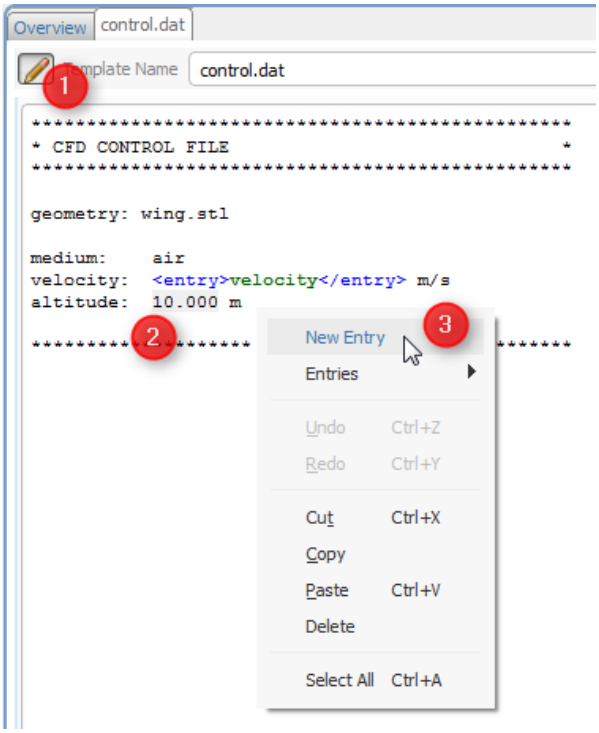
5 Template Input Files: Manual Edit

If you want to directly write into your input file (note again that this is a copy of the original file):

- Click on the edit button in the upper left corner, in order to activate the edit mode.
- Write your modifications.

This is an option if the auto-detection, from above, does not find your values:

- Replace numbers with an entry by marking them and choosing *new entry* from the context menu.



6 Result Values – Part 1

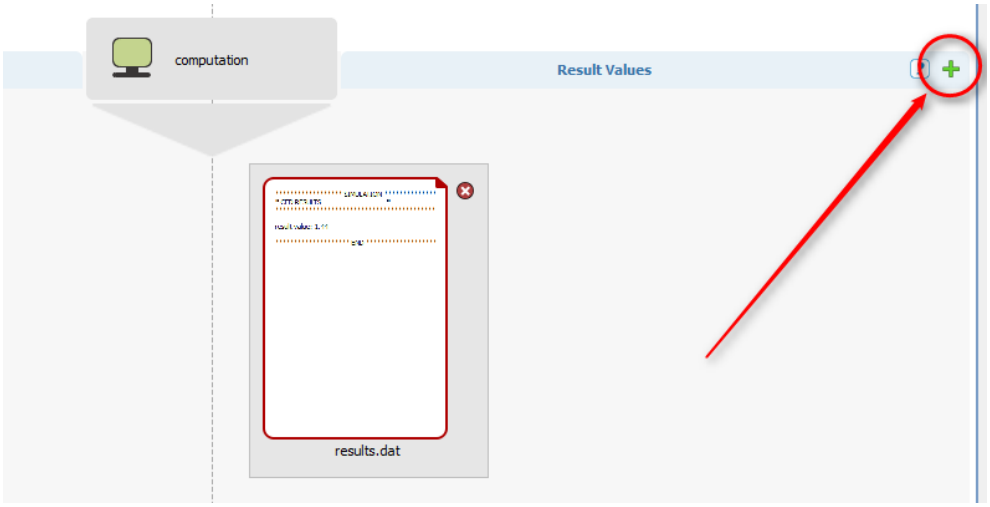
Typically, simulation tools are able to provide ASCII result data in a certain – sometimes complex – output format, for instance, showing comprehensive information for each iteration step. Such an ASCII file might contain relevant values from the analysis such as pressure loss, coefficients for lift and drag etc.

For demonstration purposes, let's assume we have such an ASCII file, called "results.dat", where our value of interest is embedded somewhere:

results.dat

```
1  ****************** SIMULATION ******************
2  * CFD RESULTS                                  *
3  ************************************************
4
5  result value: 1.44
6
7  ******************** END ************************
```

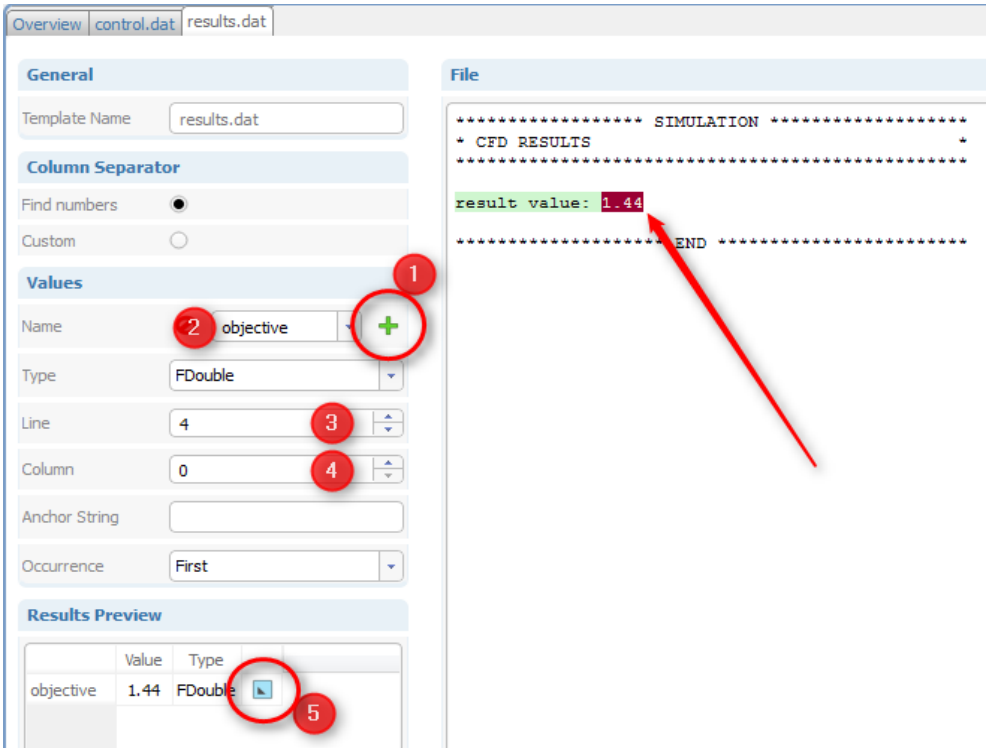
- Import a sample result file of your simulation software by clicking on the "+" button of the quadrant *Result Values*.



7

Result Values – Part 2

The values of interest will be stored in a *parameter* object so that it can be used in an automated process, such as optimization:



- Click on the corresponding tab (or double-click on the file icon of your result file) in the software connector.
- In the category *Values*, add a new item by clicking at the "+" icon and set a name i.e. identifier.
- Use the *line* and *column* editors in order to detect the value of interest. There is a preview table (category *Results Preview*) showing detected values.

✓ You can also start the line count from the end of the file by using negative values, e.g. "–1" gives you the last line of the file.

- Create a *parameter* for this value by clicking on the blue parameter icon next to the table value (point "5" in screenshot above). The parameter is created in the object tree and is now connected to this result value (i.e. based on the computation, see step 9).

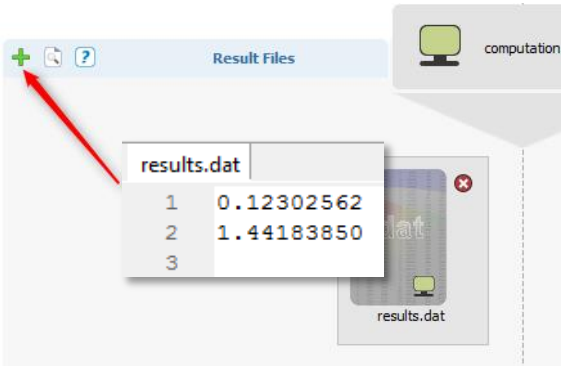
8

General Result Files

CAESES allows to read in different file formats such as png, html, vtk, case, etc. This kind of data is then simply provided in a dedicated viewer (i.e. window) for pictures and tables or even in the 3D view for interactive post-processing.

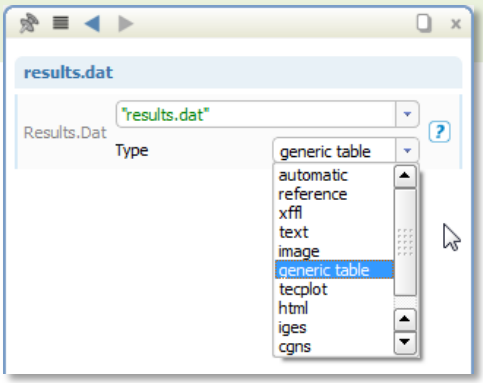
For instance, assume the simulation tool exports a file that is called "results.dat" again. But now, it contains only two values as a result of a CFD analysis.

► Select your sample result file by clicking on the "+" button of the category *Result Files*.



See step 11 which shows how this table data is provided after the run of the external software.

✓ Most of the common formats are automatically detected such as *csv* and *png* (e.g. for screenshots). However, if you have problems with this detection, try to choose from the pull-down menu of your result file.

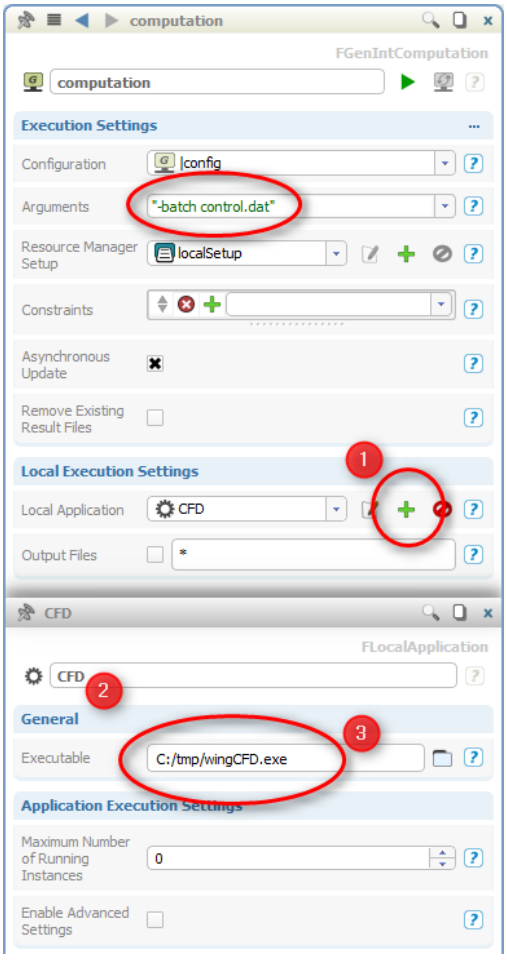


9 Computation

We still have to tell CAESES where to find the executable to the external software.

- Select the computation in the center of the software connector.
- Create a *local application* by clicking on the "+" button in the category *local execution settings*.
- Set the path to your executable, and set a name for the application.
- Set the arguments for the batch mode of the external software (note that this is optional and depends on the software).

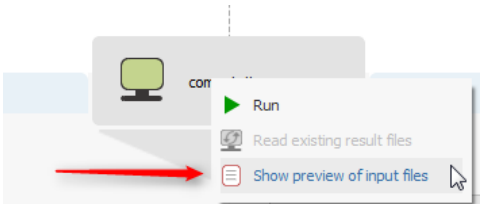
✓ If you want to skip CFD calculations when a constraint is violated, then insert the constraint directly in the computation settings, see the attribute *constraints*. This avoids expensive simulation runs if e.g. the geometry is infeasible.



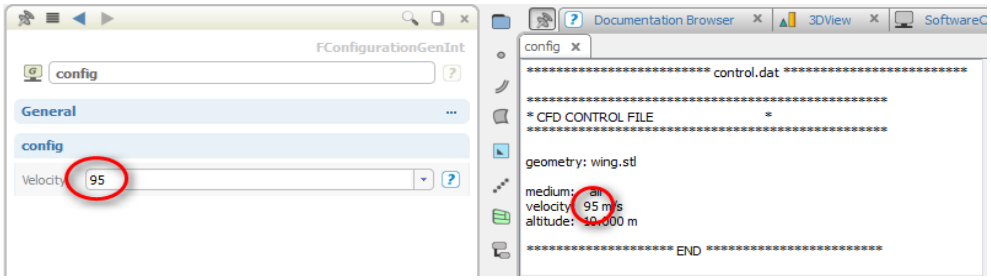
10 Check Input Files and Run

Before we start a simulation, we can have a final preview of our input file from step 4. Remember our example: The velocity will be controlled so it needs to show the same value in the generated input file.

- Choose *Show preview from Input Files* from the context menu of the computation.

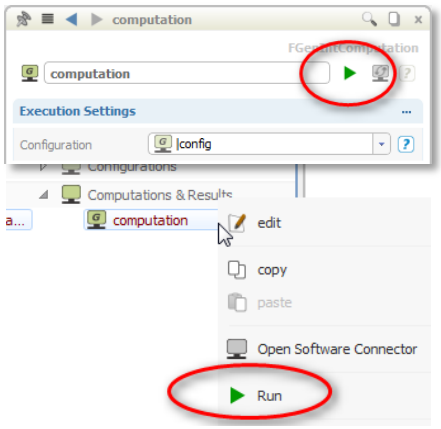


- Test: Change the value of your entry so that it gets also changed in the template:



Now, we are ready to run the external software:

- Select the computation.
- Start the computation by clicking on the run button (▶) in the upper right corner of the computation or, alternatively, use the computation's context menu either in the software connector or in the object tree (tab *Connections*).



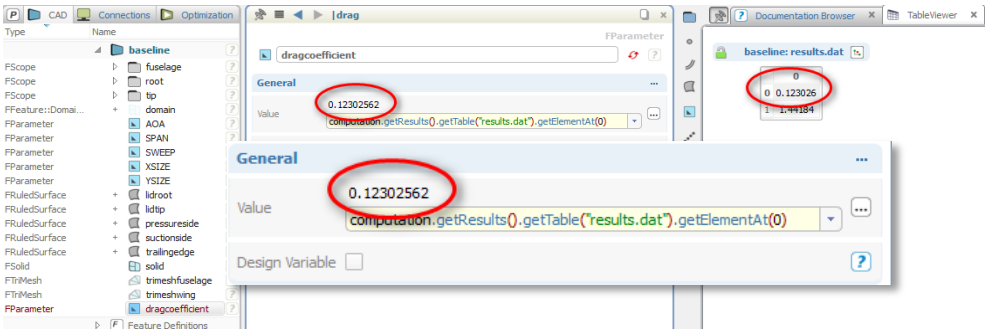
✓ Introduce *design variables* for discrete values of an input file that you want to control. Apart from geometry parameters, CFD settings can also be automatically controlled by design engines via design variables for studies such as grid studies, speed variations, etc.

11

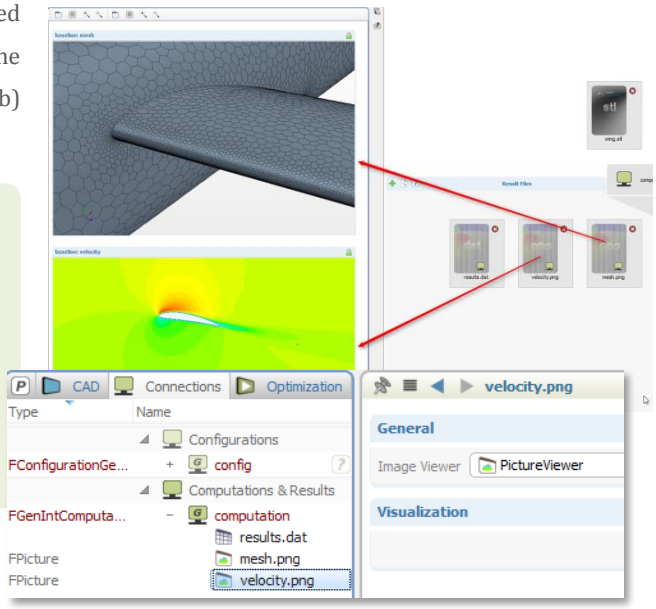
Results

When an external software run ends, all results are collected by CAESES and they are provided to the user. Result data is expected to be available in the computation directory of either the baseline design or the variants. In our case this is "<project folder>/manual_results/baseline/computation".

- Table data (e.g. the simulation results as shown in step 8) are automatically given in a table viewer. If you need a value from such a table, do the following:
- Double-click on the value in the table to create a parameter in the object tree (*CAD* tab).
- Set a name for your new parameter.



- For exported screenshots (result files with type *png*), take a look at the *picture viewer* (there is a tab given in the main window or via *view > windows > picture viewer)*. Here is an example that shows a set of screenshots:
- All result files are listed below the computation in the object tree (*Connections* tab) and can be configured.



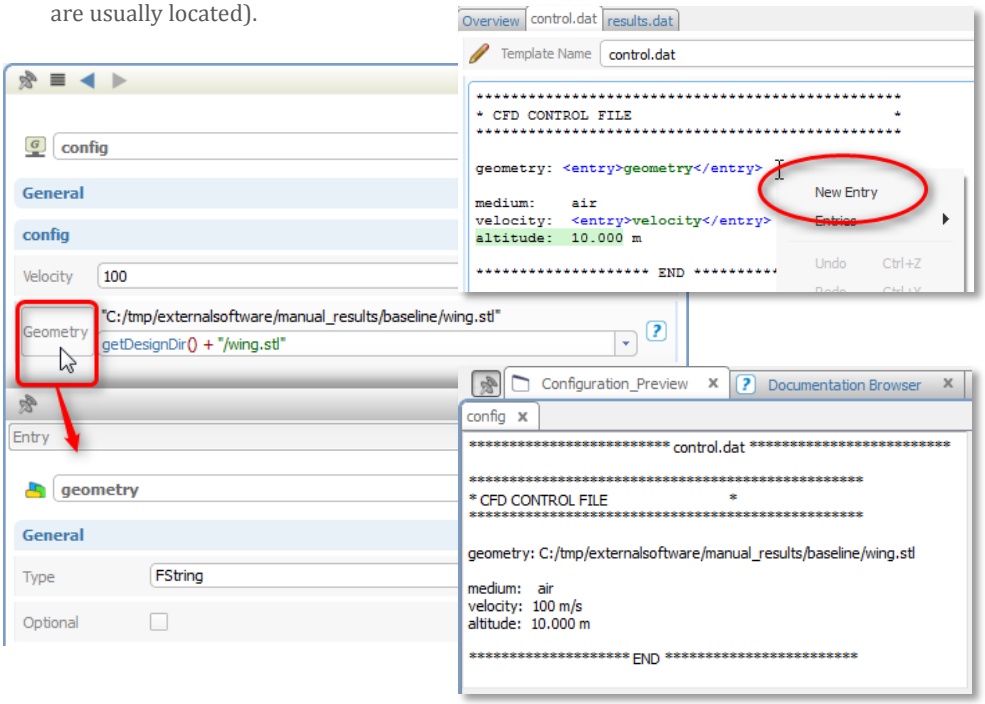
✓ Result files like *vtk, ensight gold* and *tecplot* can be directly post-processed in CAESES. Note that there is a separate tutorial for interactive post-processing of CFD data.

12

Remarks

Here are some general remarks if you want to integrate your specific simulation tool.

- If an absolute directory path needs to be replaced in an input file, add your own entry of type *FString* and use the global command *getDesignDir()*. This automatically inserts the current design directory into your ASCII file – also during runs with design engines where the design directory changes for each design (i.e. the directory where input and output files are usually located).



- For data and directory management, it is favourable if the external software automatically runs in the current working directory when triggering its batch mode. CAESES sets the working directory based on the computation and the current design.
- See step 7: Anchors can also be defined for finding a certain string occurence in a template file. Based on this anchor string, the line count is then started (instead of from the beginning of the file). The first or last occurrence of a string can also be requested. Finally, so-called *regular expressions* can be used to detect strings and values in a template file that are not detectable by the other presented options.