VS Browser Preferences

Use this screen (Figure 1) to specify and save global preferences in a VS Browser such as CarSim. It has a few default settings for VS Math Models, some global settings that control behavior of the VS Browser, and some preferences involving default datasets and external tools.
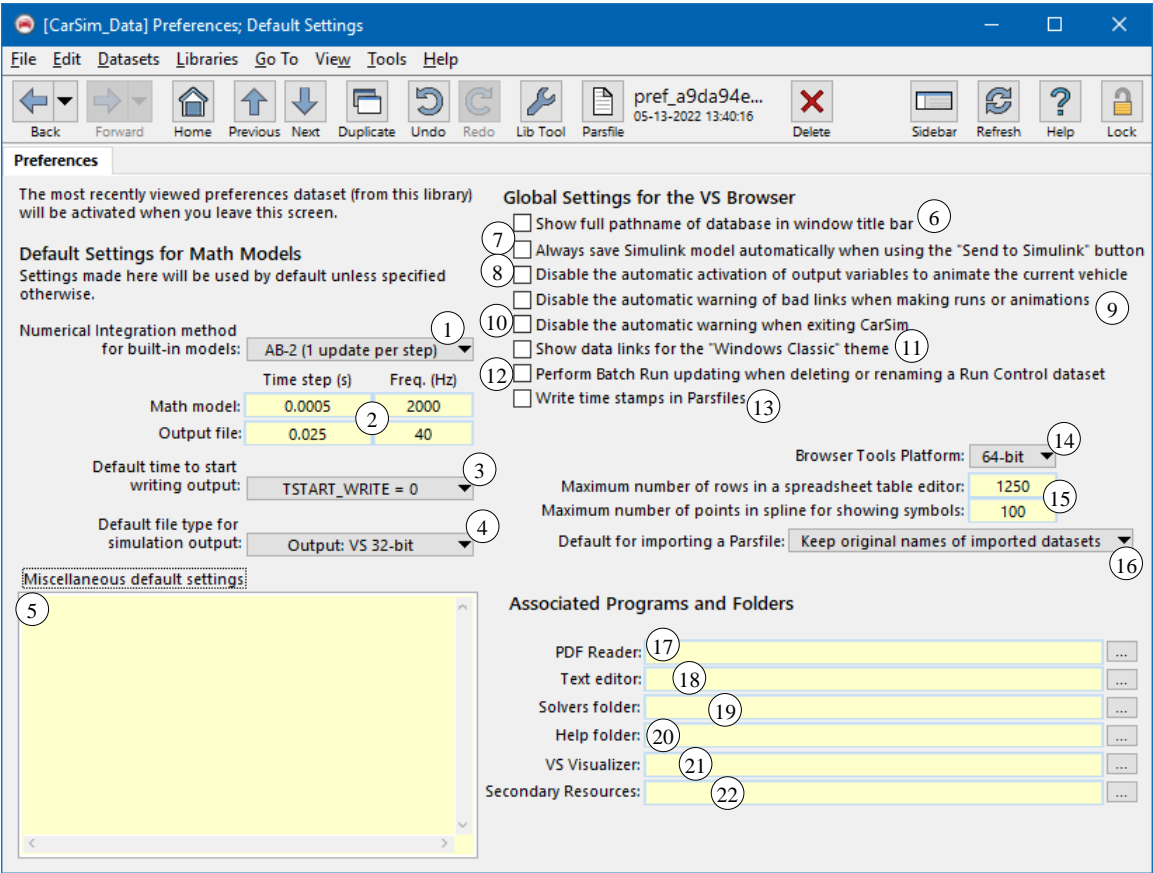

Figure 1. The Preferences screen.

The datasets associated with this screen are stored like all other datasets in the VS database, allowing you to maintain a library of preferences if you wish. Unlike most other libraries, you never link to a **Preference** dataset. To view this screen, use the **Tools** menu (the last item on the menu is **Preferences**). Once the screen is displayed, you can browse through **Preferences** datasets just as with any other library.

The key feature to understand is that you activate a set of preferences simply by viewing a dataset — the most recently viewed preferences apply, including the next time you run the VS Browser.

The screen has three regions, covering different types of settings that are described in the following three sections.

Default Settings for Math Models

Values set in the left side of the screen define default values for system parameters used in the VS Solvers. Settings made here will be used in all simulations, unless alternative values are specifically set elsewhere to override the values set here. All parameters mentioned in this section are described in more detail in the Help menu document: *System Parameters in VS Solvers*.

1. Drop-down list for selecting a numerical integration method (keyword = `OPT_INT_METHOD`). VS Solvers support six different methods for numerically integrating differential equations (Figure 2). The first two calculate model variables once per time step, and are usually set up with a time step of 0.0005s. The next four methods use a constant time step, but calculate the model variables twice per time step: once at the major step, and a second time at the half step. These are usually set up with a time step of 0.001s (which implies that the internal calculations are made at intervals of 0.0005s).
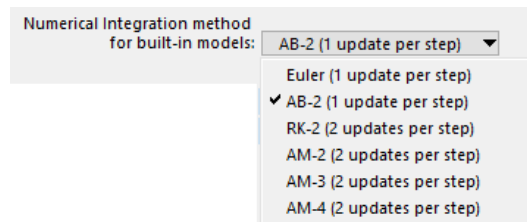

Figure 2. Options for default method of numerical integration.

The AB-2 method is recommended as the default, with the time step ② set to 0.0005s.

> **Note** Please see *VS Math Models Manual* or the technical memo *Numerical Integration in VS Solvers* for details about the numerical integration methods.

2. Control of time interval for math models and output files.

The time step can be specified either directly in seconds or as the inverse: frequency. The VS Browser automatically maintains the correct inverse relationship between these two values. If you modify the time step, the frequency is calculated and updated; if you modify the frequency, the time step is calculated and updated.

> **Note** The time step is the value passed to the math model. However, frequency may sometimes be more convenient to specify.

As with the model time step, the browser automatically maintains the correct inverse relationship between time step and frequency for the output file. It also guarantees that these

numbers are multiples of the internal time step used in the VS Math Model. Within the math model, the ratio between the time steps is specified with an integer system parameter IPRINT.

Alternate values will be used in the simulation run if set from the **Run Control** screen or a linked **Models** dataset.

> **Alert** Most parameters in the VS Solver can be specified with numbers or formulas. However, the four fields ② that specify the time step for the simulation and output file are also used to automatically calculate numbers, as described above. Because of the automatic calculations, each of these fields must contain a numerical value. Blank fields and formulas are not supported.

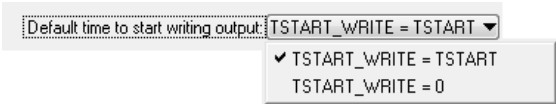③ Drop-down list for selecting a default time to start writing output (Figure 3).

Figure 3. Drop-down control for default time to start writing outputs to file.

The starting time for a run is specified as TSTART, and the time to start writing to an output file (the VS or ERD file used for saving variables that can be plotted, used for animation, or exported to other software for post-processing) is TSTART_WRITE. The two most common settings are to always start writing at T = 0, or to always start writing at the start of the run; these are the two options provided for default behavior. Alternative values or expressions can be provided for TSTART on the **Run Control** screen or any linked dataset.

④ Drop-down list for selecting a default file format for writing simulation outputs (Figure 4).
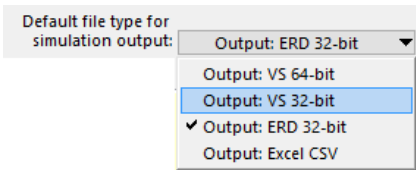

Figure 4. Drop-down control for default file format for writing simulation outputs.

VS Visualizer handles files in any of the formats.

The VS 64-bit format preserves the precision needed for GPS and possibly other variables. On the other hand, the VS 32-bit files are only half as large, and are sufficient for most variables related to the state of the vehicle (other than GPS).

Excel CSV (comma-separated variables) is a text file that can be read by most spreadsheet programs. However, it lacks auxiliary labels for variables that are used by VS Visualizer to automatically label plots with overlays of multiple datasets.

ERD 32-bit is the legacy format used prior to 2015. If you are using legacy visualization tools, it is necessary, though the legacy tools are no longer supported as of 2018.

⑤ Miscellaneous default settings. This yellow field can be used to set default values for any parameters in the VS Solvers. It is intended for setting system parameters beyond those that have already been described.

Global Settings for the VS Browser

Controls set in this section of the screen affect the way the browser communicates with the VS Solvers and other parts of the VS package, including the database.

⑥ Checkbox to show the full pathname of the database folder in the window title bar. If not checked, only the name of the folder is shown. A restart of the VS Browser is required for changes to take effect.

⑦ Checkbox to always save Simulink MDL files if they are accessed interactively from the VS Browser.

When the VS Browser is used to run a Simulink model, it sends information to set the time step and stop time. Simulink keeps this information in the MDL file, and marks the file as having been changed. When you close the MDL file, Simulink will normally prompt you to save the file.

When making a run quickly using the **Run Now** button on the **Run Control** screen or one of the batch control screens (**Batch Matrix** and **Batch Runs**), the VS Browser always sends a command to close the file while leaving it in its original state.

When viewing the model using the **Send to Simulink** button on the **Run Control** screen, the status of this box determines whether the VS Browser saves the MDL file automatically, or leaves it alone. If left alone, then Simulink will prompt you to save the file when you exit Simulink.

If you and everyone in your group at work use the same version of Simulink, then you probably want to check this box, to save the extra step of manually saving (or declining to save) the MDL file every time you view it.

On the other hand, there are situations where you do not want to automatically update the file. If the MDL file was made with an older version of MATLAB/Simulink than you are using, then when you save the file it will no longer work with the older version. If this is your situation, you should not check the box, and when prompted by Simulink to save the MDL file, do not do it.

⑧ Checkbox to disable the automatic activation of output variables to animate the current vehicle.

In normal operation, the VS Browser will scan inputs to the VS Solver to determine which output variables will be needed for the animator. These include the coordinates and angles needed to show bodies in the models (sprung mass and wheels), and possibly other animated objects such as force arrows, sensor targets, or traffic vehicles.

If you need to create small output files, you can disable this behavior by checking this box.

⑨ Checkbox to disable the automatic warning of bad links.

In normal operation, the VS Browser will scan inputs to the VS Solver. If any bad data links are found, an error message is shown and the run is not made. Check this box to disable the warning and make the run, even though some of the intended data might be missing.

(10) Checkbox to disable the automatic warning when leaving the VS Browser. The Undo/Redo system only works for changes made in the current session with the VS Browser. If you have changed anything on any screen, the VS Browser will show a popup window that asks "are you sure you want to exit?" Use this control to disable the warning if you prefer.

(11) Checkbox to show data links in the "Windows Classic" theme. Drop-down controls and blue data links are normally shown in a style compatible with Windows XP and newer. Checking this box changes the cosmetic appearance to be compatible with the "classic" styles supported by older Windows. This setting is strictly cosmetic; it has no effect on the behavior of the VS Browser.

(12) Checkbox to enable automatic updating of **Batch Runs** datasets when a change is made to a **Run Control** dataset title or category.

The **Batch Runs** screen has a list of **Run Control** datasets that can be run automatically with a single button click. If any of the **Run Control** datasets are deleted or renamed, the single click will not update perfectly. If this box is checked, all datasets in the **Batch Runs** library are automatically scanned and updated if they refer to a Run Control dataset that was deleted or renamed.

This rescanning adds a delay, so the box should be unchecked if you are not routinely running in Batch mode.

(13) Checkbox to write time stamps in expanded Parsfile used for export. In older versions (prior to 2018.0), time stamps were always written in expanded Parsfiles. This can complicate text comparisons made using version control software, so the behavior can be disabled by unchecking this box.

(14) Drop-down list to specify the default platform for VS Browser Tools. The tools include VS Visualizer, VS Scene Builder, and VS Terrain Utility. Two versions of each are installed: one is 32-bit; one is 64-bit. The 64-bit version can handle larger amounts of memory and is recommended unless you are using 32-bit Windows.

(15) Limits used for presenting tabular data; these specify a limit in the number of rows for a spreadsheet, and the maximum number of points for showing symbols in a plot with spline interpolation.

Most tables of numbers in the VS Browser can be viewed in two styles: spreadsheet or simple text. The spreadsheet style involves more formatting time, which can delay the display for very large tables. When bringing in data from older versions of the VS Browser, this value is used to determine whether the table will be shown with a spreadsheet or simple text field. If the number of rows is less than this value, the spreadsheet option is used; otherwise the simple text option is used.

When spline interpolation is selected for nonlinear tabular data, a plot is usually made showing the original data points with square symbols, and the spline interpolation shown with a solid

line. When there are many points, the symbols overlap and are not useful to see. The value determines at what size of table the symbols are no longer displayed.

(16) Drop-down list to set behavior when importing an expanded Parsfile.
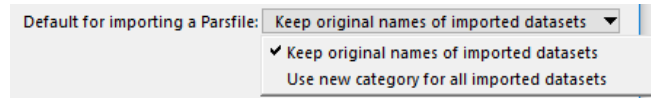
When you use the menu item **File > Import Parsfile (Any Export Type)**, there are several options for handling datasets that have the same names as datasets that already exist in your database. The two most common options are:

1. Keep the original names and categories of the imported datasets (and do not import if the combination of name and category already exists).

2. Keep the name but use a new category for the imported datasets. (Again, if a name and category already exist, the dataset is not imported.)

Use this control to specify which of the above two options should be the default. (Regardless of the setting here, all options are always available. This setting only determines which option is checked initially.)

The second method is usually undoable and is therefore recommended except when you are confident the data being imported will be kept.

Associated Programs and Folders

The files for a VehicleSim product such as CarSim are organized in two major directories: one for programs, and one for data. The program folder does not require write access, and is located by default in the Windows `Program Files` folder (typically on drive C). Most of the files in this location are not usually updated by users during normal operation of the software. This folder is usually named *product*_Prog (e.g., `CarSim2020_Prog`). This folder in turn contains a folder `Programs` where auxiliary programs are located. Most of the programs and folders in `Programs` are standard in the VehicleSim design, and must not be changed. However, a few of the tools and locations can be reset by advanced users or system administrators to support custom math models or external software.

(17) Pathname for a PDF Reader, preferably from Adobe. Nearly all of the Help documents (such as this one) are provided as PDF files in the Help folder of the product. The files have all been indexed, to support of very rapid search of the entire Help folder using the **Help > Search Help** menu item. When selected, the Browser will launch the Reader specified in this field. If the field is blank, then the default PDF Read for your Windows installation is used.

This field is provided in support of setups when the default PDF read is not from Adobe. Other PDF reader work OK for viewing the PDF files, but they do not support the **Search Help** menu option.

(18) Pathname to a text editor. This pathname identifies the program that will be used to display text files, in support of some of the VS menu items and buttons. As installed, a freeware program called ConTEXT (written by Edn Kirin) is specified. ConTEXT is configured to

show syntax coloring for Parsfiles and files with common programming formats such as C/C++.

(19) The VS Browser has a `Solvers` folder that contains Windows versions of the VS Solver library and other necessary libraries (e.g., Simulink S-Function wrapper DLLs used to link VS Solvers to Simulink). If an alternate pathname is specified here, then simulation runs made with the built-in math models will use the VS Solver DLLs from the alternate `Solvers` folder.

Be aware that the alternate folder is not used if the simulation is run under external software such as Simulink. To run with alternate VS Solver under external software, you must specify a pathname for the solver in the linked Models dataset (e.g., from the **Models: Simulink** library if running under Simulink).

If this field is blank, the browser will use the default folder location: *product*`_Prog\Programs\Solvers`.

(20) The **Help** menu for the VS Browser is generated when the browser starts, based on a text file located in the designated `Help` folder.

If this field is blank, the browser will use the default folder location: *product*`_Prog\Programs\Help`.

(21) Pathname to an alternate VS Visualizer. Setting a specific visualizer will ignore the VS Visualizer Platform option (14). Leave blank to use VS Visualizer as installed in your product.

(22) Pathname to an alternate Resources folder that the animator will use. This folder will be searched if an asset is not found in the default location. If the field is left blank, the browser will use the default folder location: *product*`_Prog\Resources`.