

SetExtFuncDllName

%SetExtFuncDllName function

Function

The function changes the dll file to be called in the execution of [external functions](#).

Declaration

```
%SetExtFuncDllName(  
    TEXT in configDllName,  
    TEXT in newDllName,  
    BOOL in global  
)
```

Parameters

configDllName	Name of the dll file defined in the configuration of object of the External function type (tab Parameters - the Dll file) parameter. If the value of the parameter is "" (i.e. empty string), there will be cancelled all the changes caused by previous calls of the function within the validity of the function. In this case, the value of the parameter <i>newDllName</i> is not taken into consideration.
newDllName	Name of a dll file to be used instead of the dll file specified by <i>configDllName</i> . If the value of the parameter is "" (i.e. empty string), there will be cancelled all the changes caused by previous calls of the function within the validity of the function.
global	Validity of the function. Possible values: <ul style="list-style-type: none">• @TRUE - global validity, i.e. the function is valid in all instances of all script within the process that executes the function (the D2000 EventHandler or D2000 HI),• @FALSE - local validity, i.e. the function is only valid in current instance of the script and is terminated after by terminating the script. Note: Local validity has a higher priority than global one.

Description

The object of the [External function](#) type represents a connection of the D2000 system with an algorithm implemented in dll file. ESL and mathematical apparatus can call (execute) those functions. Every external function has the parameter **Dll file** defined in its configuration and the parameter specifies the dll file with an external function implemented in it. Dll file usually implements several algorithms and therefore a set of external functions that shares one dll file is created in the D2000 system.

In specific cases (e.g. trying a new version of dll file) it is necessary that ESL script uses different dll file than the one specified by the configuration of external functions with no need to change the configuration. One call of the function **%SetExtFuncDllName** allows to change the dll file for all the external functions that uses the file.

The function can cancel the change of specific dll file or all dll files.

Example 1

Calling the function:

```
%SetExtFuncDllName("original.dll", "test.dll", @FALSE)
```

causes next calls of the objects of the [External function](#) types with the *original.dll* file defined in their configurations to use the *test.dll* file instead of it. The change is valid within actual script.

The following call cancels the dll file change mentioned above

```
%SetExtFuncDllName("original.dll", "", @FALSE)
```

Example 2

Calling the function:

```
%SetExtFuncDllName("original.dll", "testGlobal.dll", @TRUE)
```

causes next calls of the objects of the [External function](#) types with the *original.dll* file defined in their configurations to use the *testGlobal.dll* file instead of it. The change is valid within the process (**D2000 HI**

or **D2000 Event Handler**).

The following call cancels the dll file change mentioned above

```
%SetExtFuncDllName("original.dll", "", @TRUE)
```

Example 3

Calling the function:

```
%SetExtFuncDllName("", "testGlobal.dll", @FALSE)
```

cancels all dll file changes for all external functions that has been performed by calling the function with the parameter *global*=@FALSE within actual script.

Then calling the function:

```
%SetExtFuncDllName("", "", @TRUE)
```

cancels all dll file changes for all external functions that has been performed by calling the function with the parameter *global*=@TRUE within the process (the **D2000 HI** or **D2000 Event Handler**).



Related pages:

[ESL extended functions](#)
[Function arguments - types](#)