Date:	2019-10-17
Author:	Yann Weyland, IPG Solutions Engineering
Release No.:	CM-8.0.2

How to run TestSerie from DOS Terminal

The user often wants to automate his simulation environment and for this purpose, he generally wants not to use the CarMaker GUI but to implement some script that enables him not to use it.
This article clarifies some technical points and details how to proceed for such an automation.

Technical Background

TestSeries are objects that only the TestManager module of the CarMaker GUI can understand. The reason for this is that the CarMaker GUI is implemented in TCL scripting language and the CarMaker application that must be connected with the CarMaker GUI does not include a TCL interpreter.

Solution

This technical point nevertheless does not correspond to an impossible task to implement with CarMaker. Simply the concept differs a bit from the initial approach. The CarMaker GUI provides with the ScriptControl API (described in the programmer's guide) several commands that enable to remote control the CarMaker GUI and to automate its usage this way.

Before continuing in depth with this point, it is firstly necessary to have an interface between the DOS terminal window and the CarMaker GUI. The DOS terminal is quite primitive but it provides a scripting interface: You can implement commands within a batch file (file extension .bat) and calling this file in the terminal by entering (either its absolute or relative) path (and pressing enter obviously) is enough to trigger its execution.

This batch file give you the possibility to define temporarily environment variables and the CarMaker GUI can later access to these defined variables.

For more information regarding the supported commands in batch files and in DOS terminal, please refer to their respective documentation.

After the user finished defining some variable values in the batch script, the CarMaker GUI must be started. This application supports useful input arguments. For example, the project folder that the CarMaker GUI must select can be specified. Additionally it is possible to pass a TCL line command. This is very useful because this single line can specify to load a more complex TCL script.

This way, a TCL script can implement all the necessary commands like for example, loading and starting the convenient CarMaker application that will calculates the simulation. It can set the desired result save mode. It will load the desired TestSerie and start its execution and finally it can generate a report of it.

After this, the script closes the CarMaker application and the CarMaker GUI. In some automation contexts, it is even mandatory to close all programs related to a job in order to let the automation continues with the

next job. For this reason, the APO broker daemon should be close too because this program is started automatically by the CarMaker application if not already running on the local computer.

This program runs in background in order to let APO clients (like CarMaker GUI) find APO servers (i.e. CarMaker applications). More details on this topic in the APO documentation (help menu > additional documents > APO communication service, chapter 1.4.1).

Please find in attachment both scripts related to above description. This implementation example requires that the TCL script to be in desired project folder under Data/Script.