CANdb++ Scripts

CANdb++ Scripts are started using the function “Execute Script...” in the “File” menu.

You can also start CANdb++ Scripts using the following command line syntax:

<scriptname> <databasePath> <selectionName>

<scriptname>: name of the script executed

<databasePath>: path of the database the script is performed on

<selectionName>: name of the selected object

e.g. „add_canoe_attributes motbus.dbc“

calc_busload.vbs communication.mdc PowerTrain

Return Values

CANdb++ Scripts return one of the following values:

0x00: Script has been executed successfully

0x01: Script hasn’t been executed successfully

0x02: Script has changed the database, reload database in CANdb++ Editor

These values may be combined bitwise.

Add_CANgen_Attributes.exe

Adds the following attribute definitions for CAN driver generators to an existing database:

Name	Objecttype	Valuetype	Default
MessageManagerUsed	Node	Integer	0
GenMsgCycleTime	Message	Integer	0
GenMsgCycleTimeFast	Message	Integer	0
GenMsgDelayTime	Message	Integer	0
GenMsgNoIalSupport	Message	Integer	0
GenMsgNrOfRepitition	Message	Integer	0
GenMsgStartDelayTime	Message	Integer	0
GenSigInactiveValue	Signal	Integer	0
GenSigSendType	Signal	Integer	0

Add_CANoe_Attributes.exe

Adds the following attribute definitions for CANoe simulations to an existing database:

Name	Objecttype	Valuetype	Default
CANoeStartDelay	Node	Integer	0

CANoeDrift	Node	Integer	0
CANoeJitterMin	Node	Integer	0
CANoeJitterMax	Node	Integer	0

Add_CAPLGen_Basic_Attributes.exe

Adds the following basic attribute definitions for CAPL generator tool to an existing database:

Name	Objecttype	Valuetype	Default
GenNWMApBusSleep	Network	String	“apBusSleep()”
GenNWMApCanNormal	Network	String	“apCanNormal()”
GenNWMApCanOff	Network	String	“apCanOff()”
GenNWMApCanOn	Network	String	“apCanOn()”
GenNWMApCanSleep	Network	String	“apCanSleep()”
GenNWMApCanWakeUp	Network	String	ApCanWakeUp()”
GenNWMGotoMode_Awake	Network	String	“GotoMode_Awake()”
GenNWMGotoMode_BusSleep	Network	String	“GotoMode_BusSleep()”
GenNWMSleepTime	Network	Integer	0”
GenNWMTalkNM	Network	String	“TalkNM()”
NodeLayerModules	Node	String	“”
GenNodSleepTime	Node	Integer	0
NWM-Knoten	Node	Enum	nein
GenMsgDelayTime	Message	Integer	1
GenMsgNrOfRepetitions	Message	Integer	1
GenMsgSendType	Message	Enum	spontaneous
NWM-Botschaft	Message	Enum	nein
GenMsgCycleTime	Message	Integer	100
GenSigInactiveValue	Signal	Integer	0
GenSigSendType	Signal	Enum	
GenSigStartValue	Signal	Float	0
NWM-WakeupAllowed	Signal	Enum	

Add_CAPLGen_Extended_Attributes.exe

Adds the basic set of attribute definitions and the following set of extended definitions for CAPL generator tool to an existing database:

Name	Objecttype	Valuetype	Default
GenEnvVarPrefix	Network	String	“Env”
GenEnvVarEndingSnd	Network	String	“_”

GenEnvVarEndingDsp	Network	String	“Dsp_”
GenNodAutoGenSnd	Node	Enum	
GenMsgPreSetting	Message	String	“”
GenMsgPreIfSetting	Message	String	“”
GenMsgPostSetting	Message	String	“”
GenMsgPostIfSetting	Message	String	“”
GenMsgEVName	Message	String	“”
GenMsgConditionalSend	Message	String	“”
GenMsgAutoGenSnd	Message	Enum	
GenMsgAutoGenDsp	Message	Enum	
GenMsgAltSetting	Message	String	“”
GenSigReceiveSetting	Signal	String	“”
GenSigPreSetting	Signal	String	“”
GenSigPreIfSetting	Signal	String	“”
GenSigEnvVarType	Signal	Enum	undef
GenSigConditionalSend	Signal	String	“”
GenSigAutoGenSnd	Signal	Enum	
GenSigAutoGenDsp	Signal	Enum	
GenSigAssignSetting	Signal	String	“”
GenSigAltSetting	Signal	String	“”
GenSigPostSetting	Signal	String	“”
GenSigPostIfSetting	Signal	String	“”
GenSigEVName	Signal	String	“”
GenEnvAutoGenCtrl	Envvar	Enum	No
GenEnvMsgOffset	Envvar	Integer	0
GenEnvMsgName	Envvar	String	“”
GenEnvControlType	Envvar	Enum	

Copy_Attributes.exe

Copies the attribute definitions of a source database into a destination database. The path to the destination database may be provided as first command line argument.

Calc_Busload.vbs

Calculates the busload of the selected network. The name of the network must be provided as the second command line argument. The default values used in this script must be changed accordingly.

Check_Gateway_Signals.vbs

Checks the gateway signals of a control unit. Gateway signals must have the value for attribute 'Gatewaysignal' (case sensitive) set. An RX-Signal and a TX-Signal representing a gateway signal in a control unit, must have the same value for this attribute.

Add_Gateway_Attributes.vbs

Adds the attribute ‘Gatewaysignals’ for checking the gateway signals with script ‘check_gateway_signals.vbs’.

MakeTxRxObjects.vbs

For each mapped Tx or Rx signal defined for a node, this script adds an unmapped Rx or Tx signal to a node.

Databases created with version 1.0 of CANdb++ Admin don’t support the relationship between unmapped Tx and Rx signals and nodes. To convert an existing CANdb++ 1.0 database to the new format, you have to open the MDC database and export it with the function "Export – In CANdb++ Database" in the "File"-Menu.

ChangeSigNames.vbs

This script shows the possibilities to change the names or other attributes of the objects of a certain kind in a complete database. In this example a suffix “_CAN” is appended to all signals in the opened database.