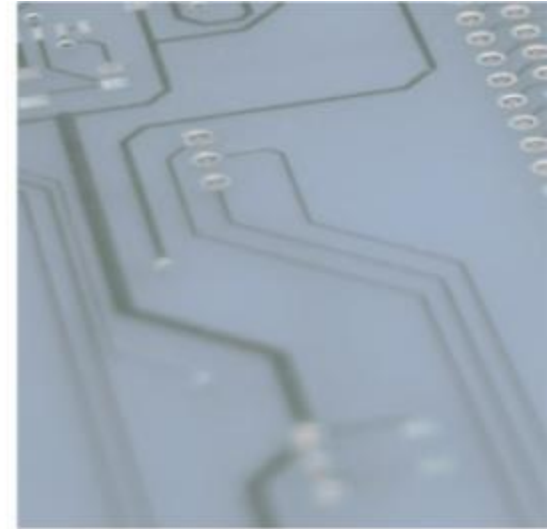
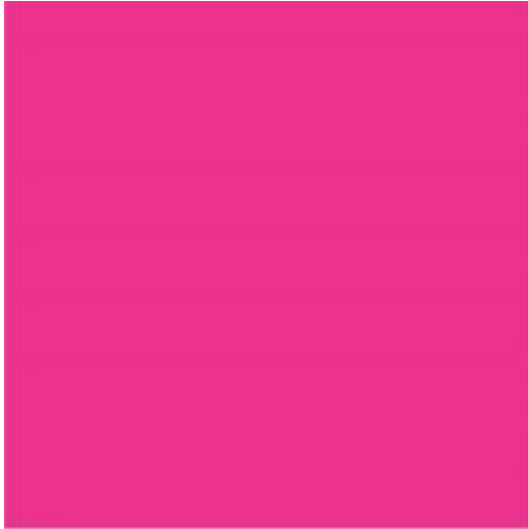


# CanEasy – Use cases



# CanEasy – Use cases

## – Motivation –

- CanEasy is not just a residual bus simulation for the development of ECUs
- CanEasy is the solution for many applications and can be extended very quickly to meet new requirements

# CanEasy – Use cases

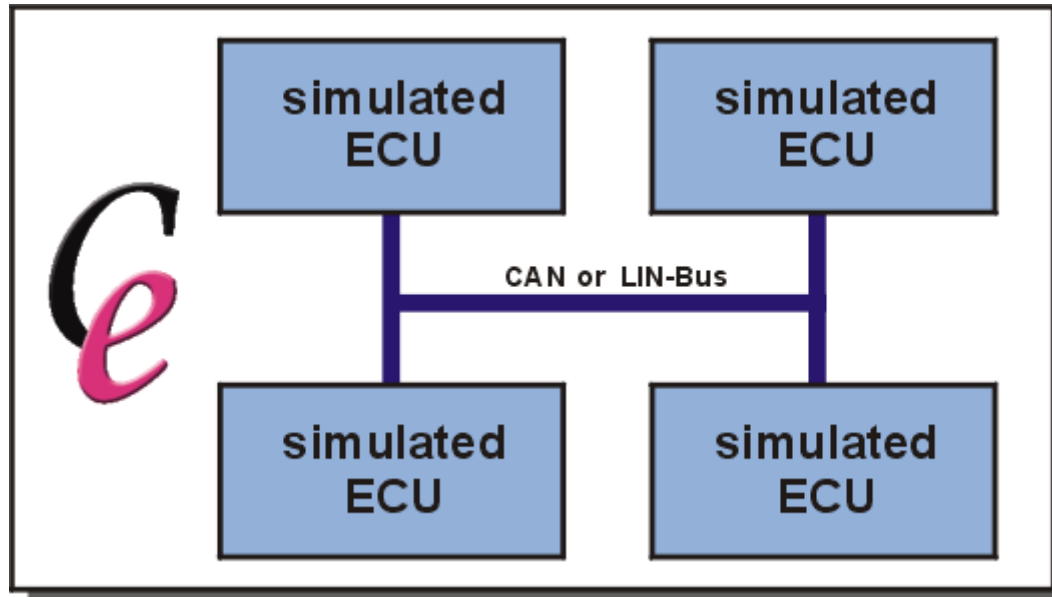
## – Overview –

CanEasy as

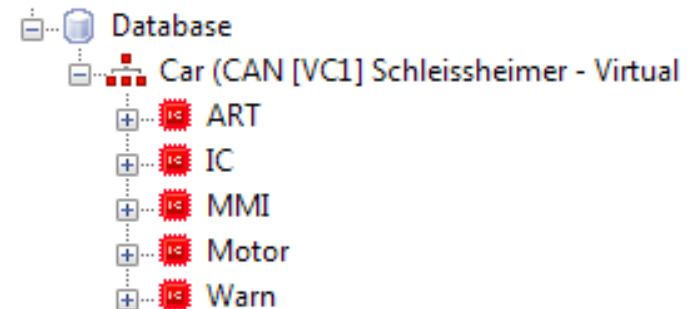
- **Virtual system**  
[all control units are simulated]
- **Residual bus simulation**  
[missing control units are simulated]
- **Analyze real network**  
[all control units are real]
- **Database editor**  
[Import, modify and export a communication matrix]
- **Test sequencer**  
[Using VBA/VSTA or schedule tables, test sequences can be automated]
- **Hardware abstraction layer**  
[Integration into another process to have a uniform access to bus adapters]

# CanEasy - Use cases

## - Virtual system -

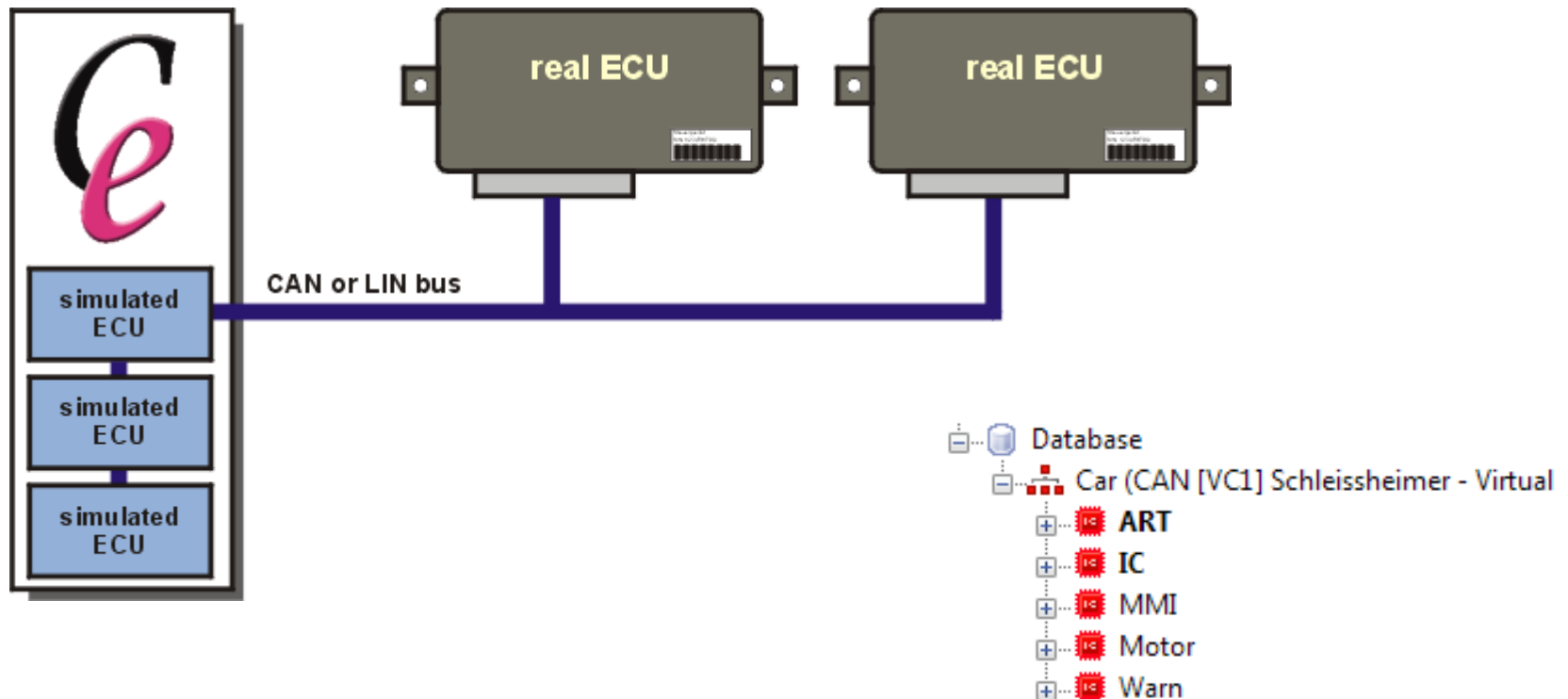


- All control units are simulated
- You can implement and test ECU functions without any hardware



# CanEasy - Use cases

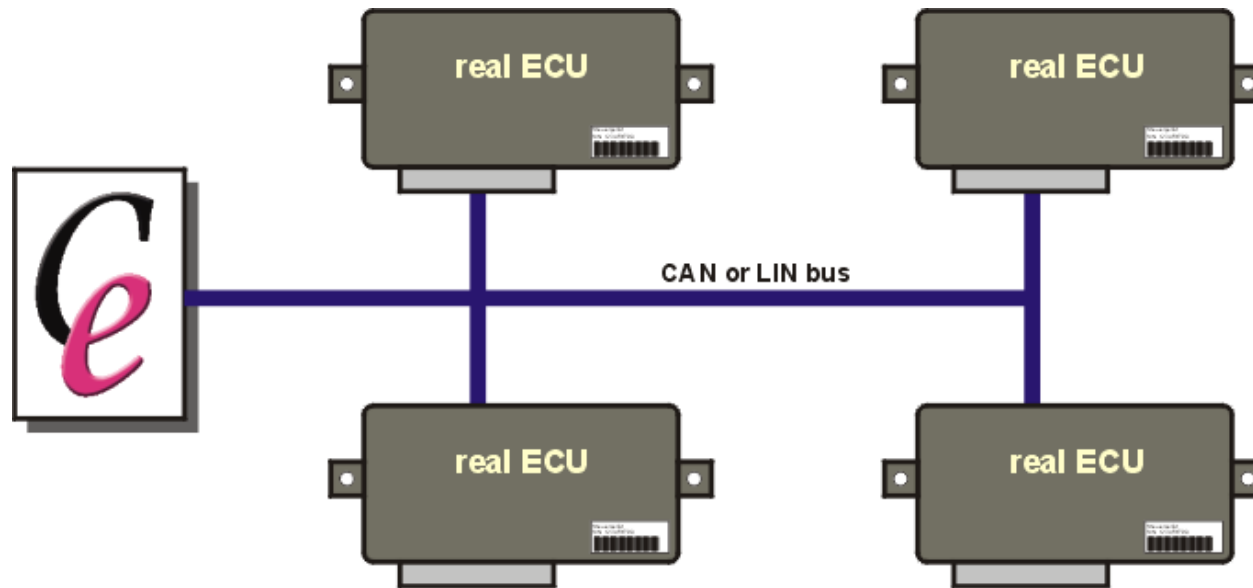
## - Residual bus simulation -



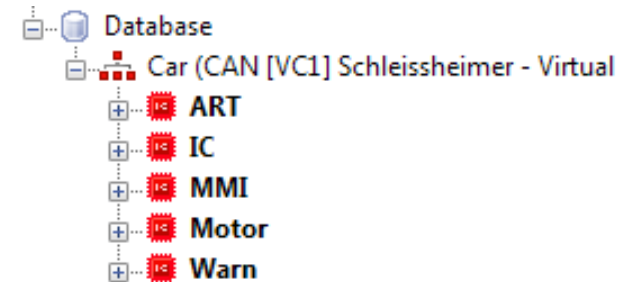
- Missing ECUs are simulated automatically
- Change the simulation behavior depending on received messages

# CanEasy – Use cases

## – Analyse real network –



- All ECUs are real
- CanEasy is just listening and records the complete bus traffic
- Use plot and trace to analyze communication



# CanEasy – Use cases

## – Database editor –

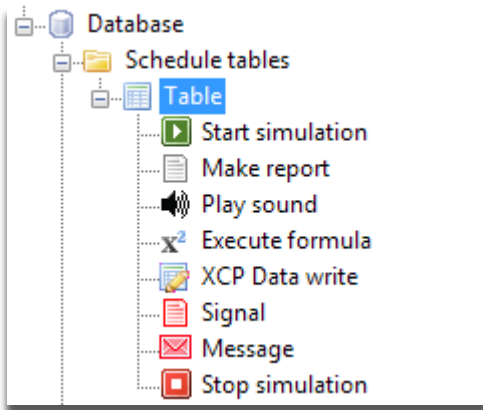
The screenshot displays the CanEasy Database Editor interface with several windows open:

- Message editor:** Shows a table with columns Name, Identifier, Cycle time, and Description. The entry 'Assistance1' has Identifier '0x190' and Cycle time '100'.
- Signal editor:** Shows a table with columns Name, Description, Byte order, Startbit, Length, and M. The entry 'CarDistance' is selected, showing Description 'Entfernung Fahr...', Startbit '24', and Length '08'.
- Received signals from...:** A tree view showing the CAN bus structure with nodes like ART, Assistance1 [0x190], MMI, DisplayProperties [0x0C8], Motor, Motor1 [0x12C], Motor2 [0x12D], and various sensors like CurrentSpeed, RPM, Tanklevel, CoolantTemp, OilTemp, and OutsideTemp.
- Hex editor 'Assistanc...':** Displays a hex dump of data: 61 00 66 00 65 a.f.e, 00 74 00 64 00 .t.d., 00 00 00 ...
- Value editor:** Shows the configuration for the 'CarDetected' signal, including Name, Description, Data type (UChar), Display type (Checkbox), Value type (Dezimal), and Start value (checked).

- Import some external communication matrix to see content in CanEasy
- Modify messages, signals, variables, ...
- Export to external file (e.g. DBC)

# CanEasy – Use cases

## – Test Sequenzer –



Option Explicit

Sub TestSequence()

StartSimulation

Call MakeReport("Test started", ReportTypeInfoInformation)

Dim oSig As Signal

Set oSig = Database.GetObjectByStringRef("Sig:Drehzahl")

oSig.ValuePhys = 120

oSig.Send

Sleep (500)

oSig.ValuePhys = 200

oSig.Send

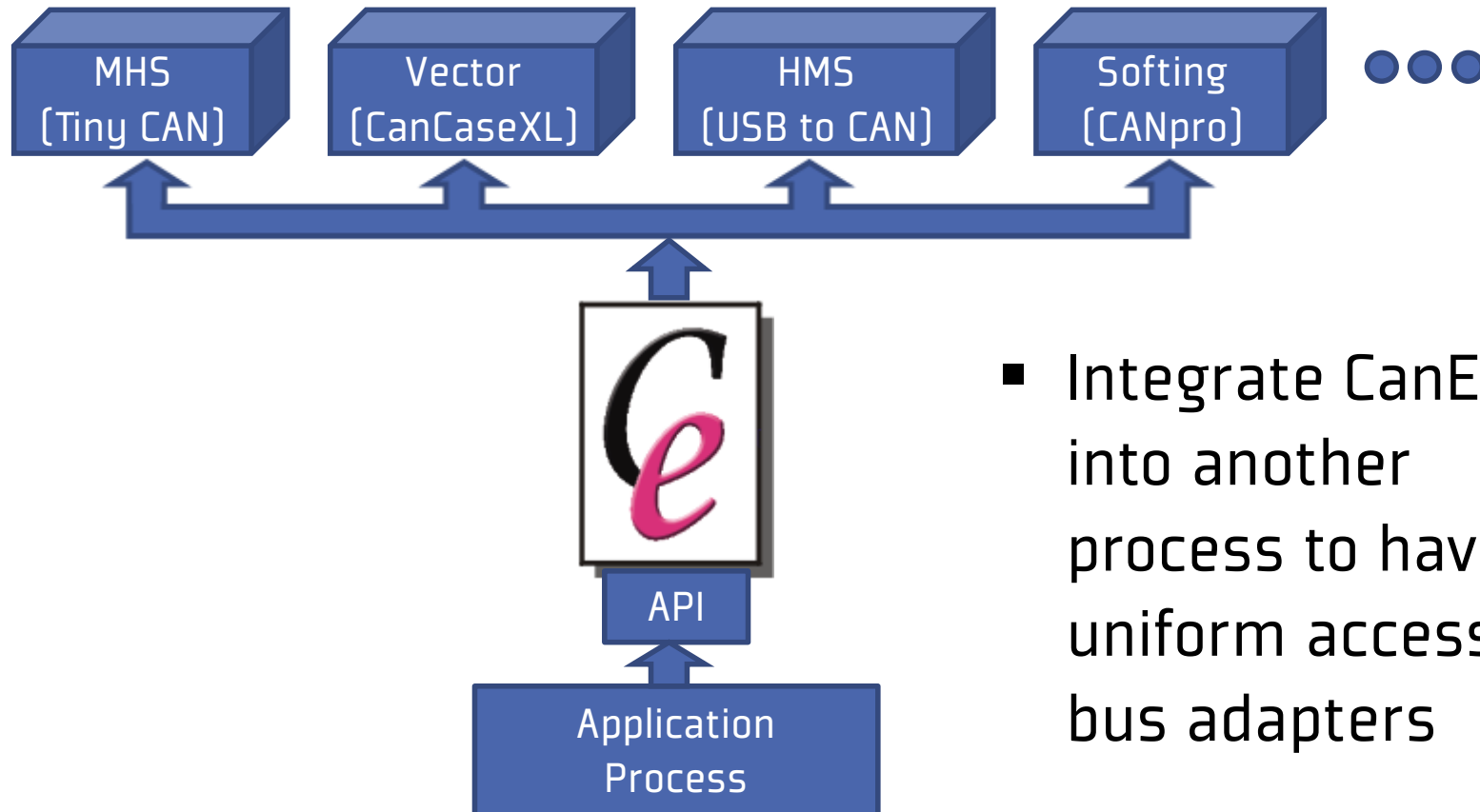
StopSimulation

End Sub

- Using VBA/VSTA or schedule tables, test sequences can be automated

# CanEasy – Use cases

## – Hardware abstraction layer –



- Integrate CanEasy into another process to have a uniform access to bus adapters

Thank you for your attention!

---