KULI 5.0

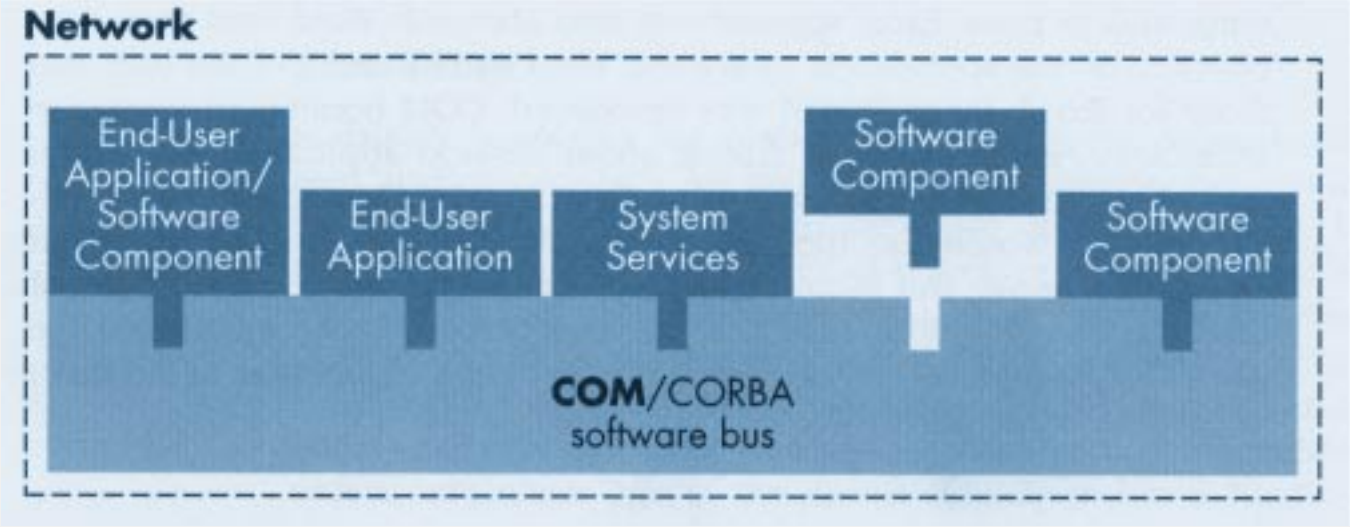
E. Geierlehner

ECS Steyr

The Component Object Model

Microsoft COM
Component Object Model

Also known as ActiveX®, OLE
Extensions: DCOM, COM+,MTS, Windows® DNA



Advantages: Programming Language Independence
Location Transparency
The Operating System provides Runtime Environment

KULI as COM Server

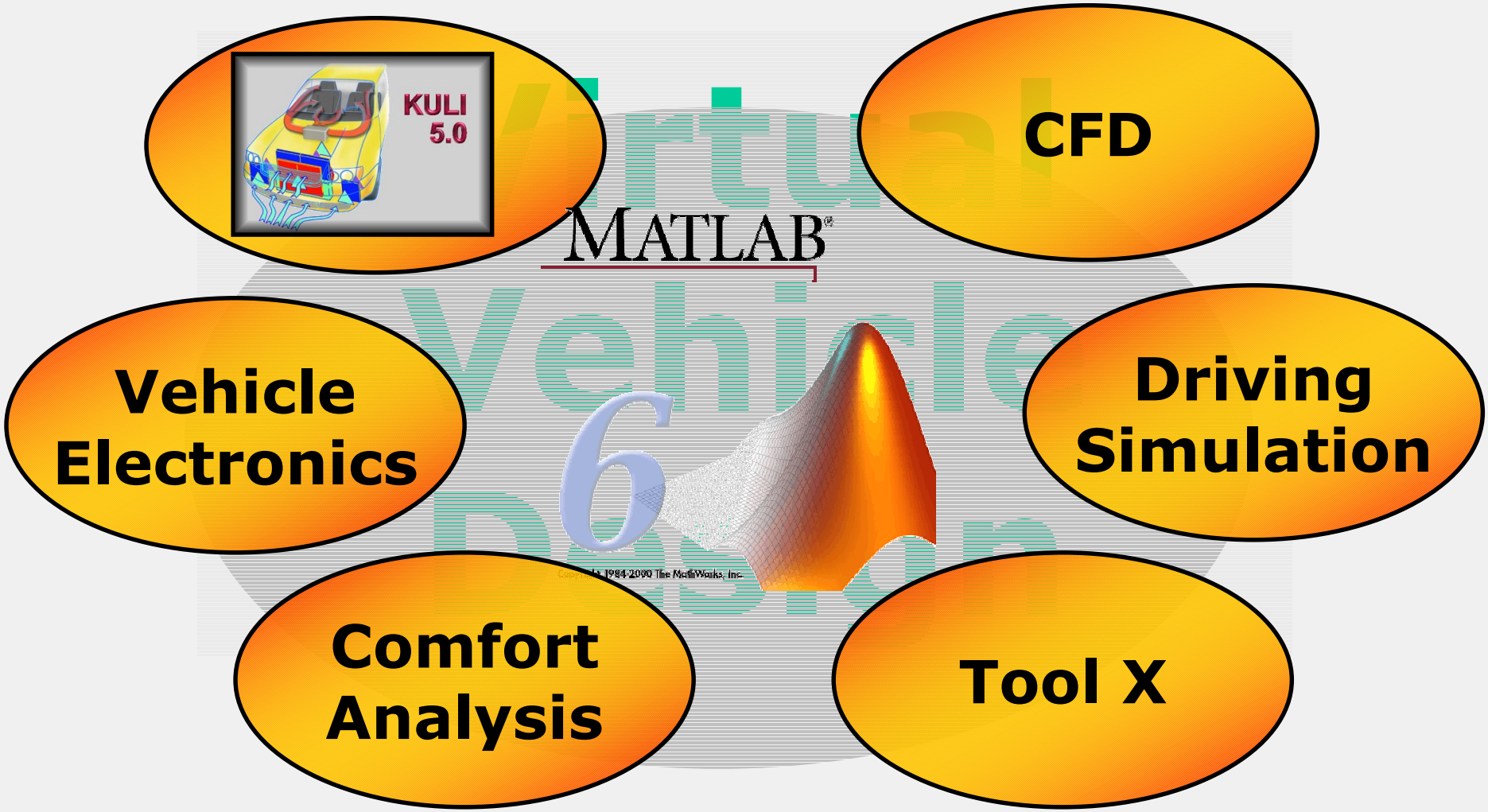
Motivation

- Make KULI accessible for Application Frameworks
- Enhance existing Interfaces
- Combine the Comfort of the KULI GUI with the Power of System Integration Tools like Matlab®

Benefits

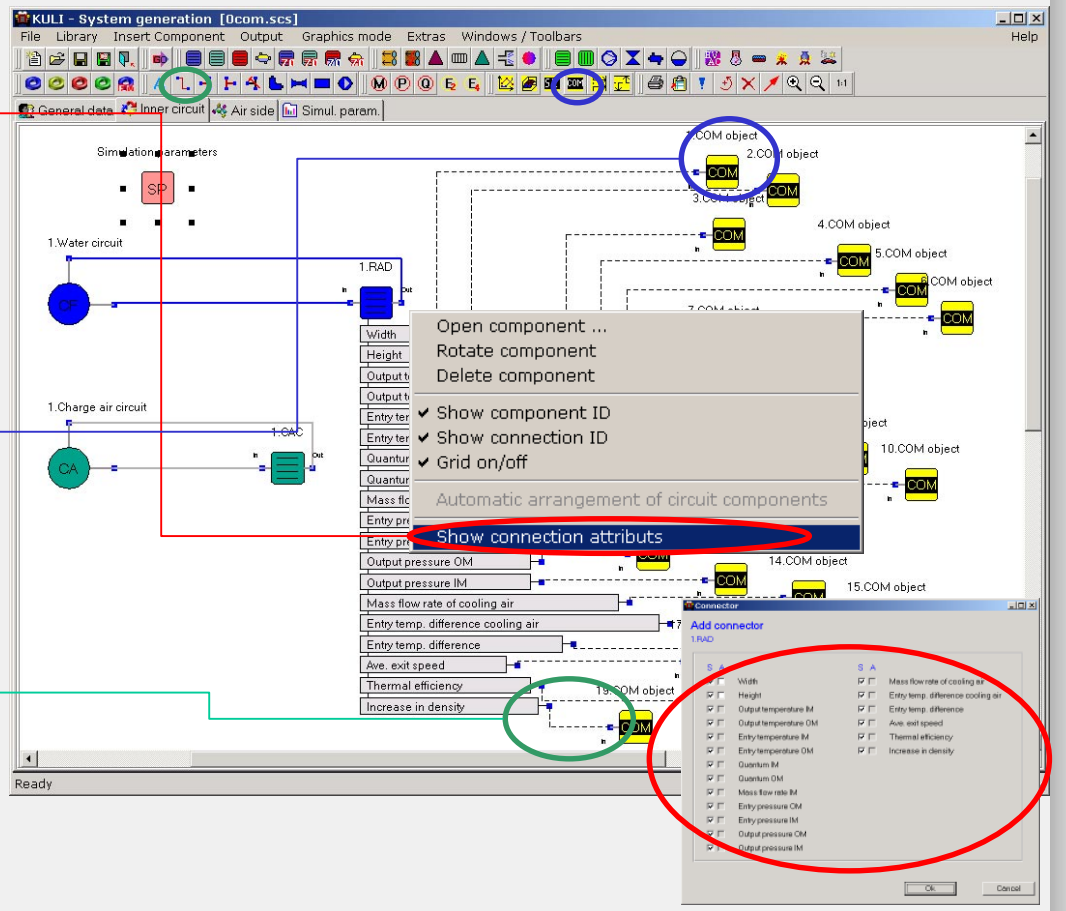
- KULI is callable from VBA and Matlab
- Integration of KULI in Expert Systems
- Integration of External Components into a KULI Model
- Free programmable external Controllers (Fans, Water, Pumps, Thermostats, ...)

Analysis Tools Integration



Using KULI's COM Component- Part 1

- **Add Connectors**
  - e.g. Entry temp. Air
  - Mass flow
- **Add COM Components**
  - e.g. COM1
  - COM2
- **Connect Connectors with COM Components**



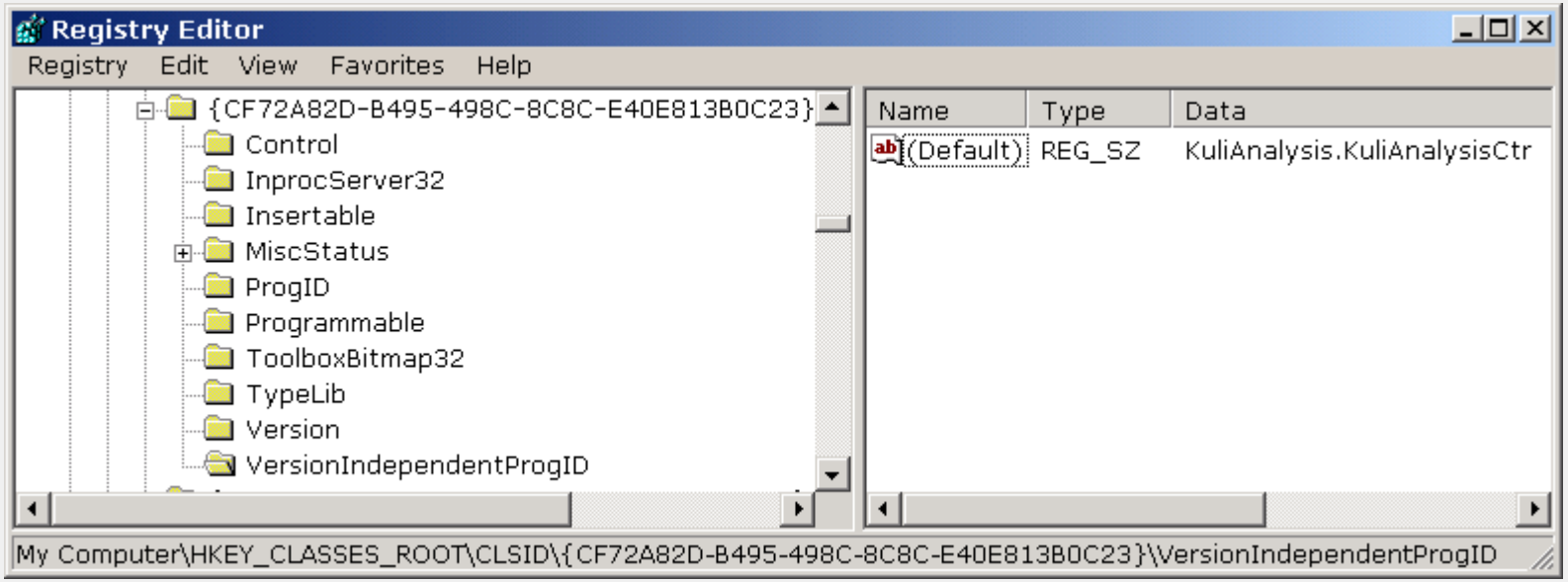
Definition

KULI's GUI COM Component is NO Software Component

COM

It is a KULI Component, which is accessible by the Interface Functions

Get / SetCOMValueByID defined in KuliAnalysis



Using KULI's COM Component- Part 2

- **COM Components are as easy to use as other KULI Comps**
  - + Benefit from KULI GUI when using COM objects
  - + No "Programming Expert Knowledge" required
- **All Information is available in KULI's Graph Window**
  - + No components, connectors, COM objects, etc are hidden
- **COM Objects as Watch-Variable**
  - + "Debugging" of unstable Systems
  - + Runtime Visualization of Parameter changes in VBA / Matlab / COM supporting Tool
- **User friendly Interface Design**
  - + Few Entry Points and Events
- **Accessing COM Objects via user definable Texts**
  - + GetComValueByID("ExitTempAtRadiator1")

Interface Functions - Events

➢ Methods / Properties

- IKuliAnalysisCtr
  - AddToBatchList(BSTR fileName)
  - BatchMode(BOOL * pVal)
  - Cancel()
  - CleanUp(BOOL * succ)
  - EnableEvents(BOOL newVal)
  - EnableEvents(BOOL * pVal)
  - GetCOMValueByID(BSTR comName, double * value)
  - Initialize(BOOL * succ)
  - IsFinished(BOOL * val)
  - IsNextOperatingPoint(BOOL * val)
  - IsNextTimeStep(BOOL * val)
  - KuliFileName(BSTR newVal)
  - NextKULIIteration(BOOL * succ)
  - PPFileName(BSTR * pVal)
  - ResultFileName(BSTR * pVal)
  - RunAnalysis(BOOL * succ)
  - SetCOMValueByID(BSTR comName, double value, BOOL * succ)
  - ShowResult()
  - StartAnalysis(BOOL * succ)

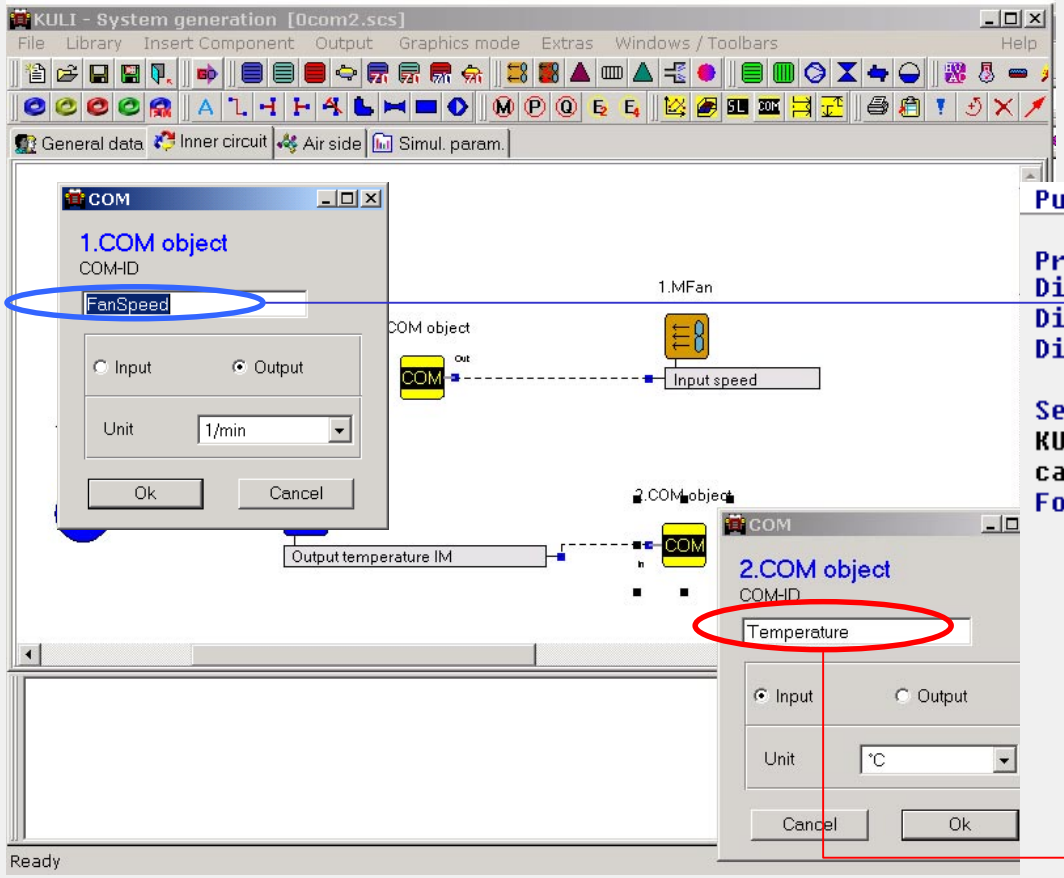
➢ Events

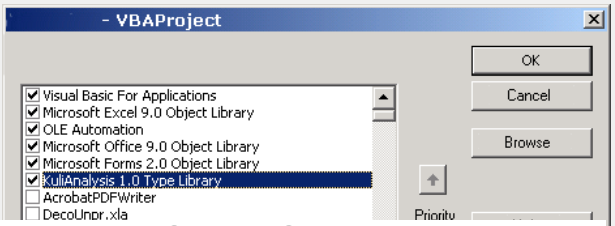
- _IKuliAnalysisCtrEvents
  - OnCheckForCancel()
  - OnError(BSTR fkt, BSTR msg, BSTR add, long type)
  - OnMessage(BSTR fkt, BSTR msg, BSTR add, long type)
  - OnNextIteration(long itNo)
  - OnNextTime(long timeStepNo, double time)

Implementation Details

- **KULI Analysis is implemented as "In-Process-Server" (dll)**
  - + Optimum Runtime Performance
  - + A few Interface Functions leads to Simple Handling
- **Direct Matlab® / Simulink® Integration in KULI**
  - + Simulink Controllers are available as Icons
  - + C++ Calls to Matlab / Simulink -> Runtime Performance
- **KULI GUI is not accessible via COM**
  - + All Required Features are implemented in KULI GUI
  - + Benefit of the user friendly KULI GUI when creating Models

Call KULI from VBA (Excel®)






```
Public WithEvents KULI As KuliAnalysisCtr

Private Sub CommandButton1_Click()
Dim value As Double
Dim rpm As Double
Dim i As Integer

Set KULI = New KuliAnalysisCtr
KULI.KuliFileName = "...\ExTruck_50_T_COMSet.scs"
calcOK = KULI.Initialize()
For j = 1 To 5
    calcOK = KULI.StartAnalysis()

    drehzahl = j * 1000
    calcOK = KULI.SetCOMValueByID("FanSpeed", rpm)
    Cells(1, j) = rpm
    i = 2
    Do Until KULI.IsFinished
        calcOK = KULI.NextKULIIteration()
        value = KULI.GetCOMValueByID("Temperature")
        Cells(i, j) = value
        i = i + 1
    Loop
        i = i + 1
    Loop
Next j

Set KULI = Nothing
End Sub
```

Usage of Get/SetCOMValueByID

- **KULI Analysis should be started using StartAnalysis()**

  Note: RunAnalysis() does not stop after an iteration

Syntax SetCOMValueByID("**FanSpeed**",**1234**)

+ Set only possible after StartAnalysis()
  and NextKULIIteration()

+ Type of COM object must be "Output"

Syntax GetCOMValueByID("**Temperature**")

+ Get can be called at any time
+ Type of COM object must be "Input"

Handling Events

- **KULI fires 5 Events**
  - OnError / OnMessage
  - OnNextIteration
  - OnNextTime
  - OnCheckForCancel
- **Only "informing" Events, no Requests**
- **EnableEvents must be set to TRUE**

KULI | OnCheckForCancel

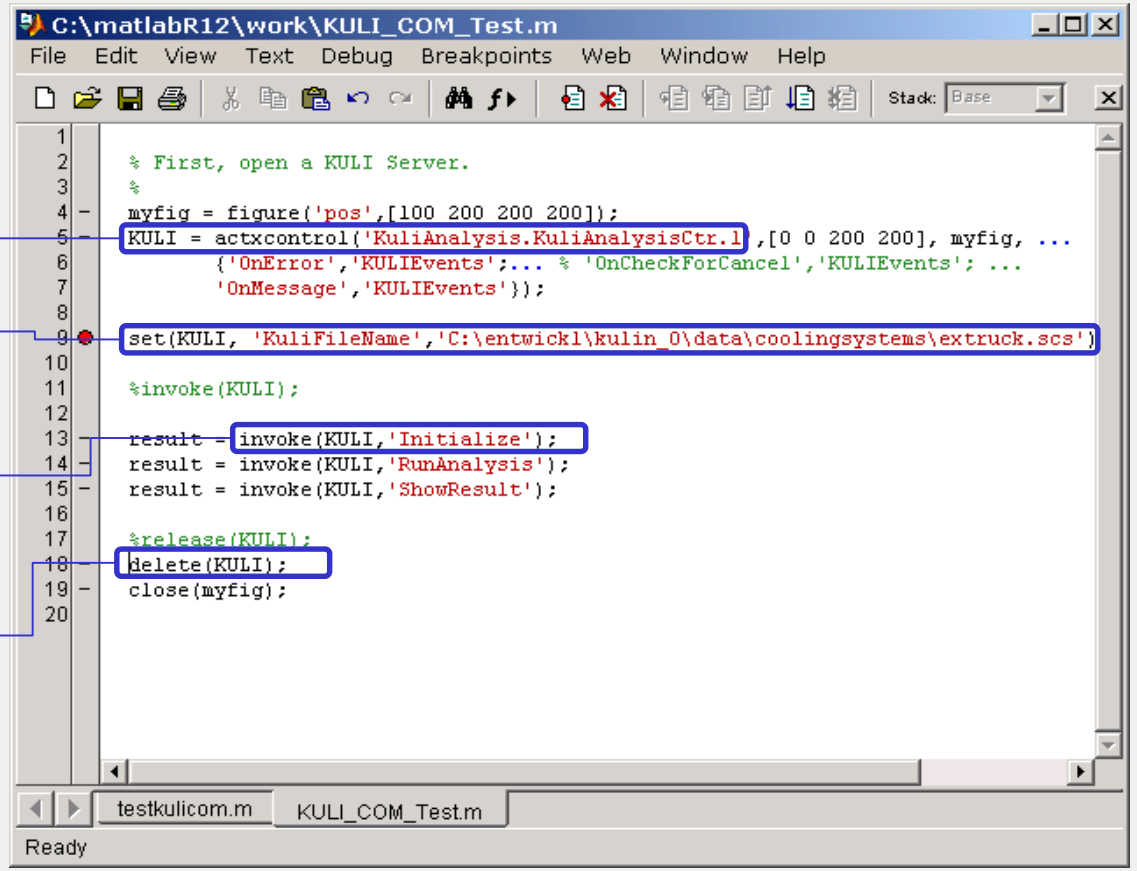
```
Private Sub KULI_OnCheckForCancel()
Dim Condition As Boolean

Condition = True
....
If (Condition) Then
   KULI.Cancel
End If

End Sub
```

```
Private Sub KULI_OnError(ByVal fkt As String, _
                         ByVal msg As String, _
                         ByVal add As String, _
                         ByVal xtype As Long)

Debug.Print fkt, msg, add

End Sub
```

```
Private Sub KULI_OnNextIteration(ByVal itNo As Long)

Dim value As Double

value = KULI.GetCOMValueByID("Temperature")

End Sub
```

Call KULI from Matlab

- Create ActiveX Control
- Set System File Name
- Run KULI
- Free Control



External Components Integration

➢ 2 Models available

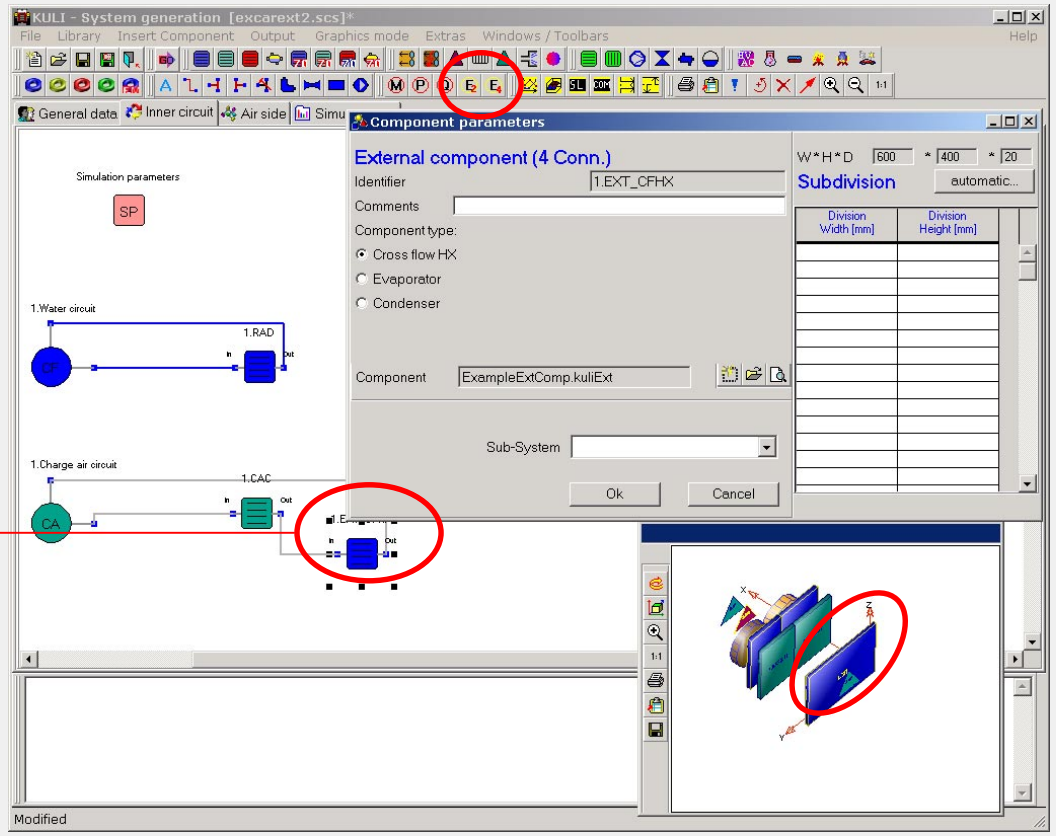
- 2-Arm External Components (Heat Exchangers)
- 4-Arm External Components (Parallel Flow Heat Exchangers)

➢ Full integration in KULI GUI

- External Components are handled like common KULI components

➢ Full Features User definable Analysis Methods

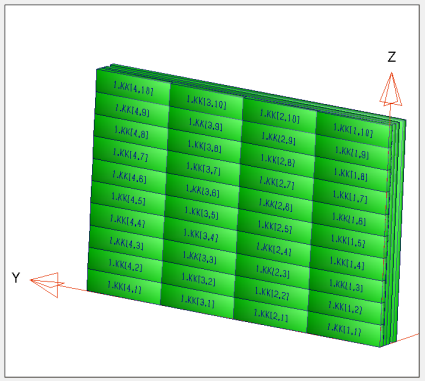
- End Users can provide their own models



External Components Specifications

- **User Defined COM Component called "KuliExtComp" providing the methods:**

  `InitializeComponent`
  `SetComponentInlet_Inside  SetComponentInlet_Outside`
  `GetComponentOutlet_Inside GetComponentOutlet_Outside`

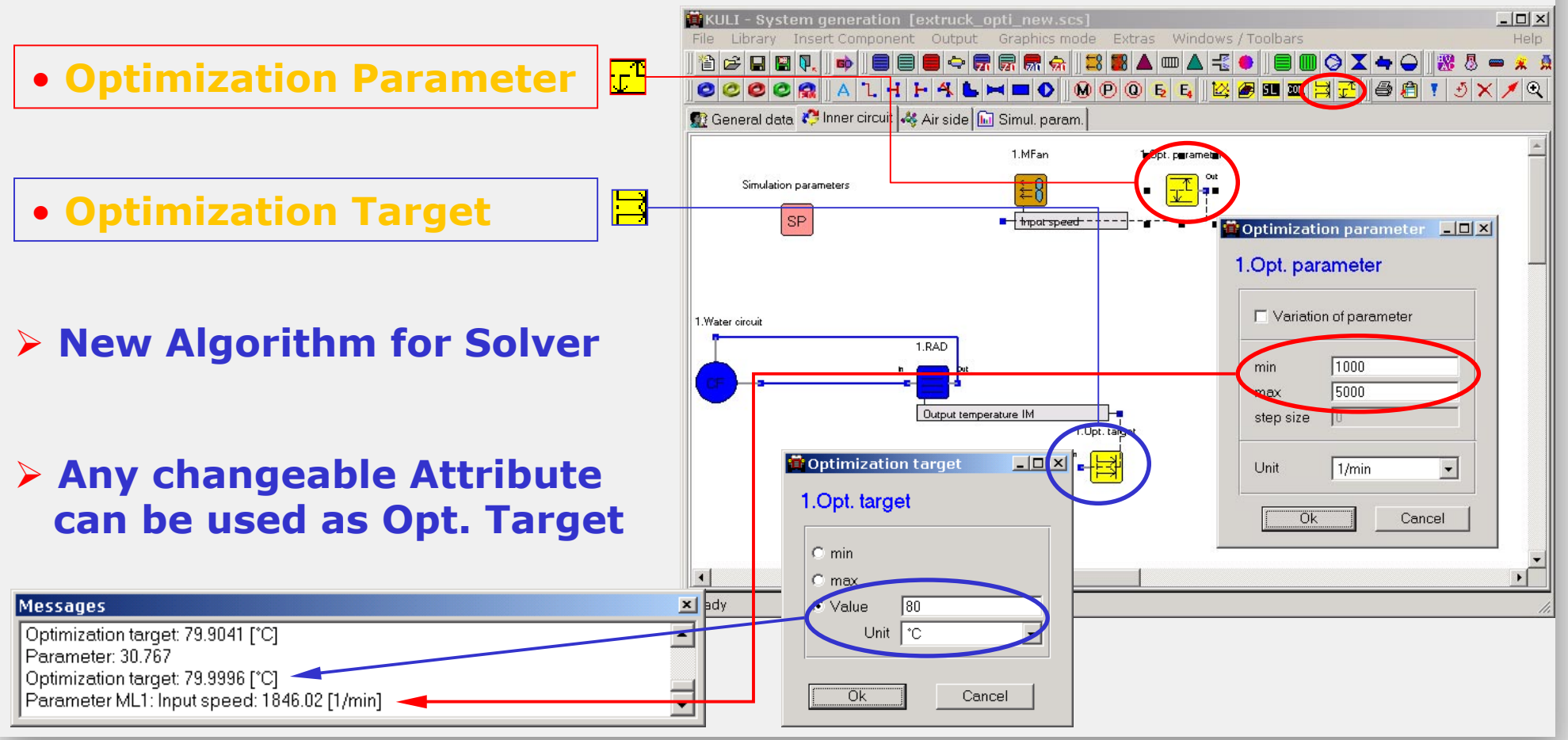


- **"KuliExtComp" can be generated with any COM supporting Tool**
- **Component overlapping not considered. Only mean values for temperature, pressure and an overall value for the mass flow rate are passed**
- **Information about the cooling media (air and fluid properties) is hard to interchange. The user is responsible for equal properties in both programs.**
- **KULI does not have access to any geometric information of the „User Defined" component**

```
library KuliExtComp
{
 InitializeComponent        (BSTR kuli_ID, BSTR filename,
                             BSTR comment,
                             VARIANT_BOOL is_in_AC_Circuit);

 SetComponentInlet_Inside   (BSTR kuli_ID,  double temp_In,
                             double press_In,
                             double x_In,    double massflow);

 SetComponentInlet_Outside ( BSTR kuli_ID,   double temp_In,
                             double press_In,  double x_In,
                           | double massflow);

 GetComponentOutlet_Inside ( BSTR kuli_ID,  double* temp_Out,
                             double* press_Out,   double* x_Out);

 GetComponentOutlet_Outside( BSTR kuli_ID,  double* temp_Out,
                             double* press_Out,  double* x_Out);
};
```

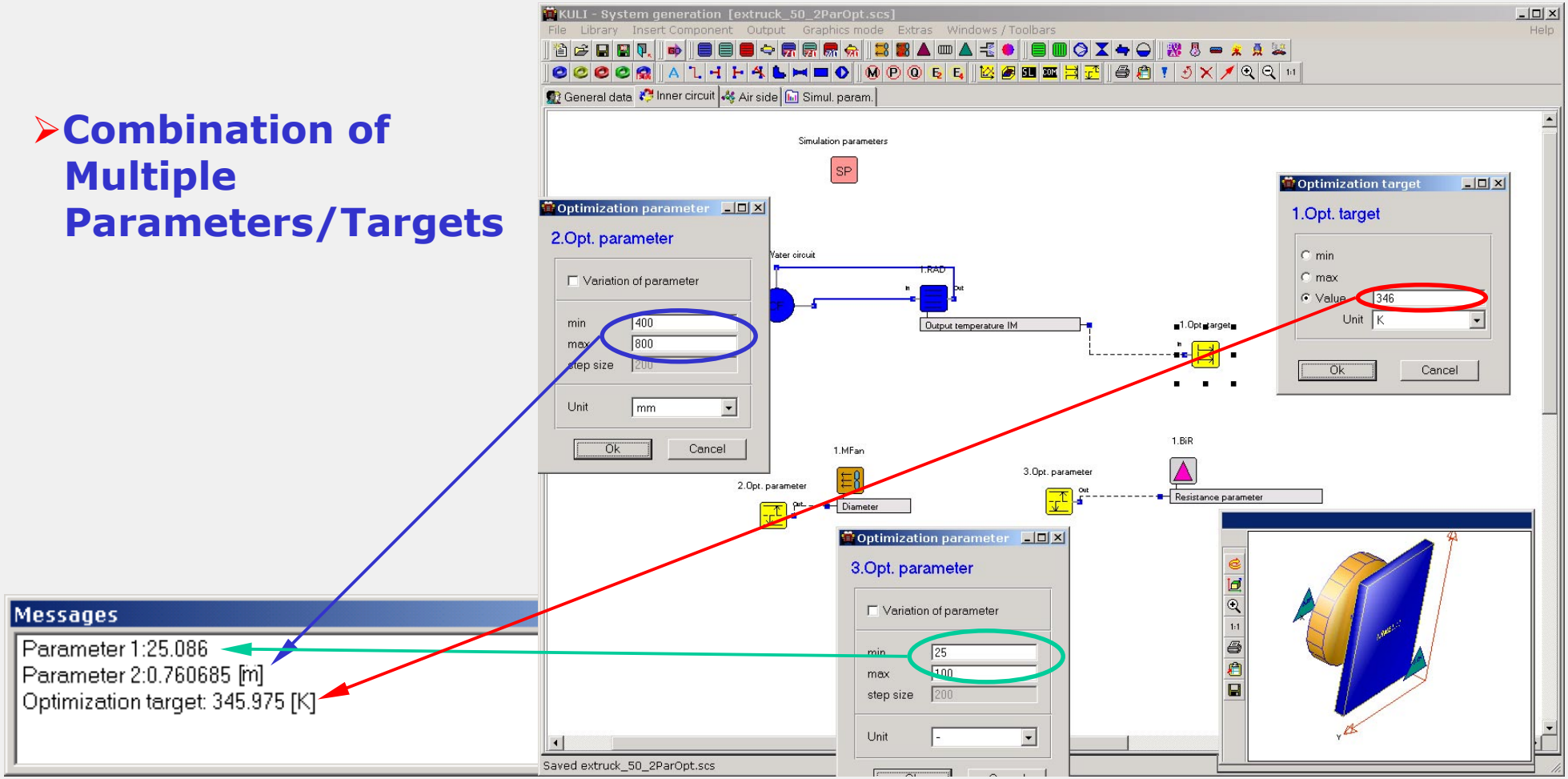
New KULI Optimization

- **New KULI Components**
  - **Optimization Parameter**
  - **Optimization Target**
- **New Algorithm for Solver**
- **Any changeable Attribute can be used as Opt. Target**



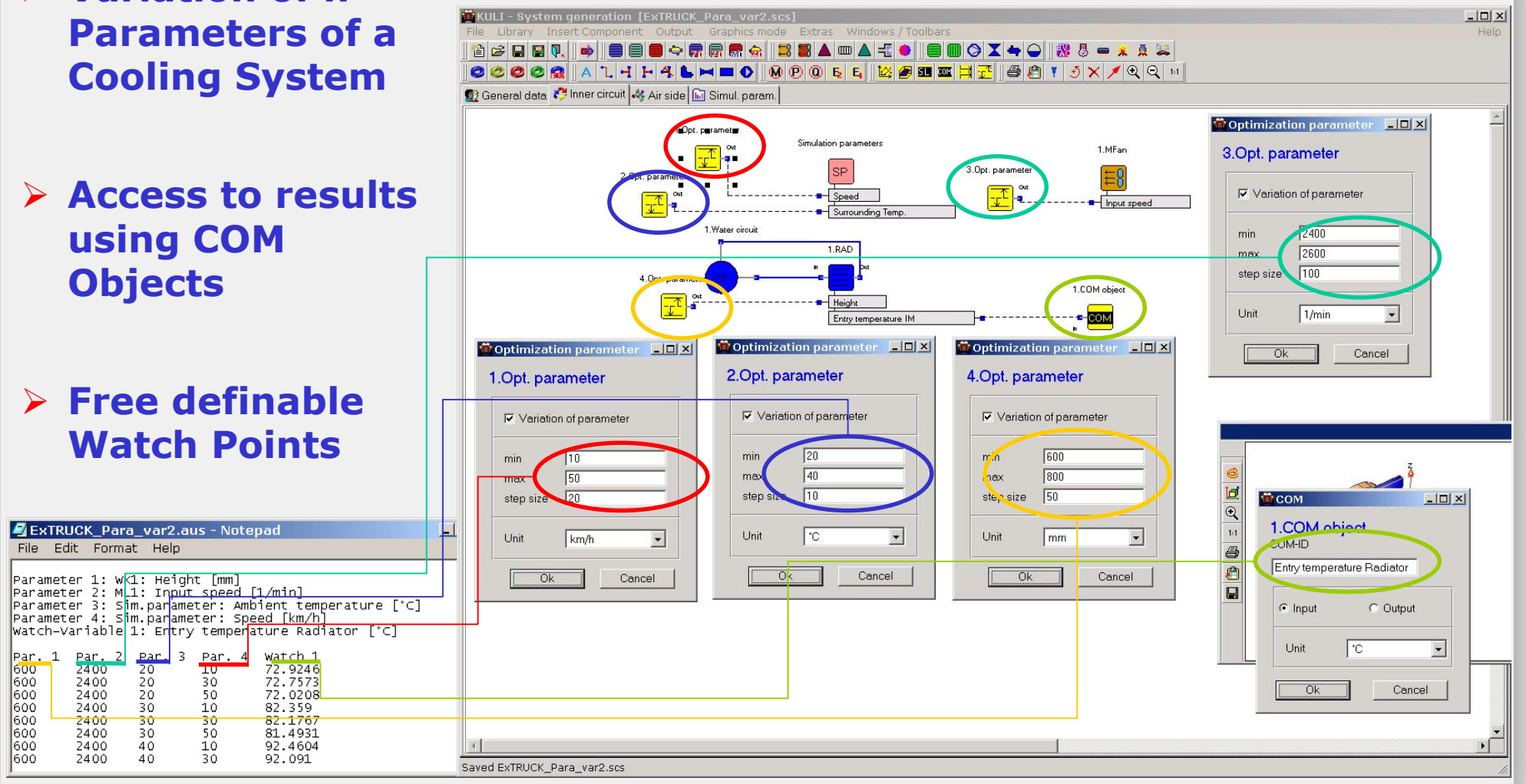
Multi Parameter Optimization

- **Combination of Multiple Parameters/Targets**



Variation of Parameters

- **Variation of n Parameters of a Cooling System**
- **Access to results using COM Objects**
- **Free definable Watch Points**



Licensing

Module BASE:

Limitations:

- No Support for External Components
- No Support for GetCOMValueByID, SetCOMValueByID

Module Advanced:

- ✓ Full Support of all Features

Future Perspectives

External Components Enhancements

- Support for "Rastering"
- Additional Components (Air Side)

Enhancements with CFD Interface

- "Interactive" CFD Interface additional to CFD Preprocessor

Post processing

- Full Support for KULI's COM Components
- Enhanced Output Options for Optimization / Parameter Variation