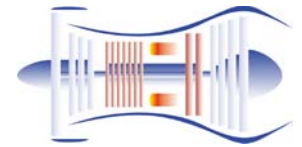
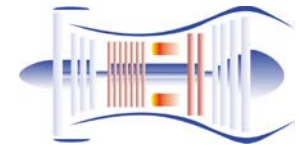


INTRODUCTION

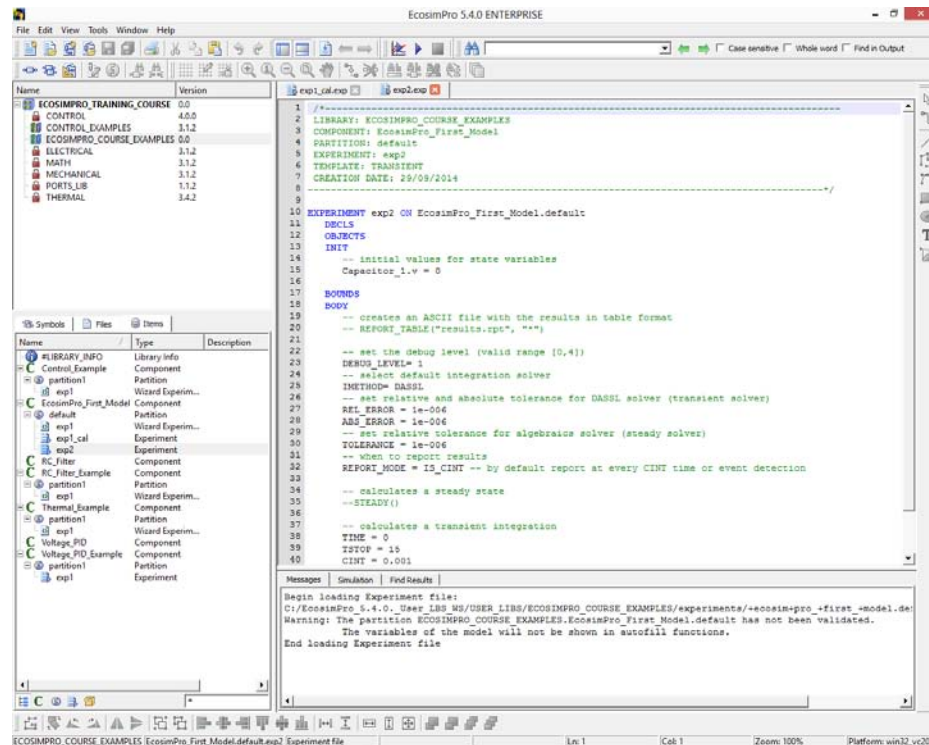


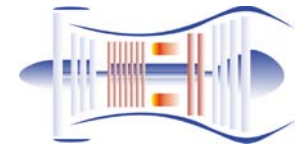
PROOSIS

- Simple and complex physical processes modeling software.
- EcosimPro Language **EL** to model continuous and discrete processes.
- Differential Algebraic Equations (**DAEs**), Ordinary Differential Equations (**ODEs**) and Discrete Events solver.
- EcosimPro provides a set of reusable **Libraries**: Mechanical, Electrical, Thermal, Mathematical, Control,...
- **Schematic** Modeling: user-friendly environment.

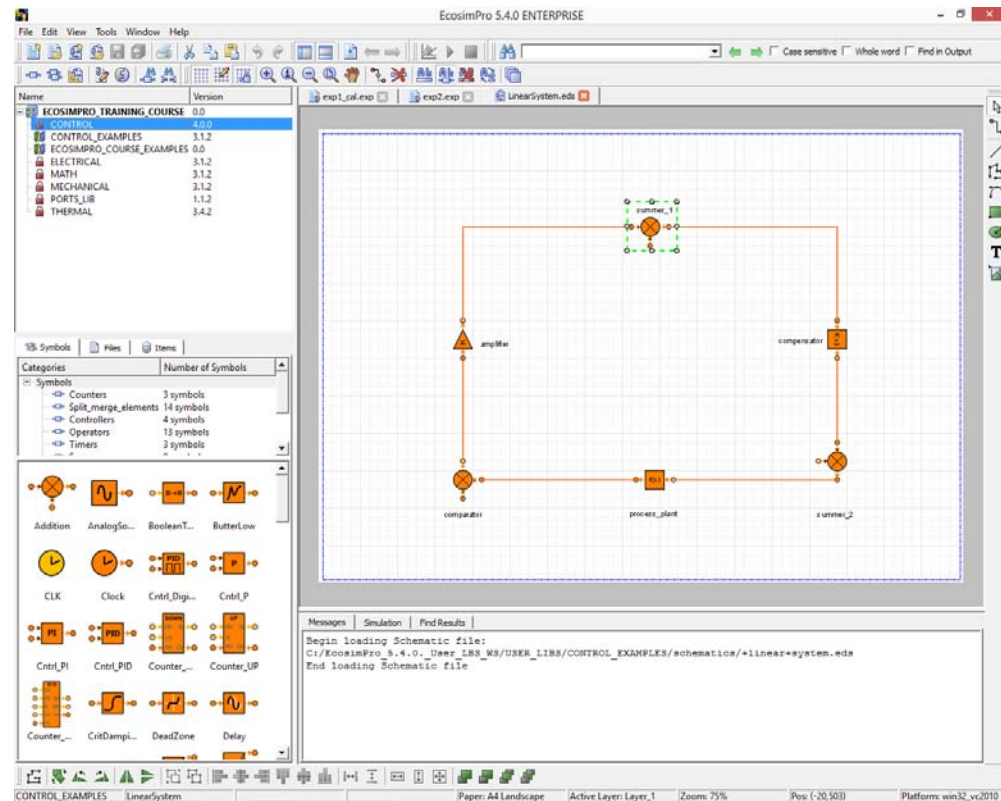


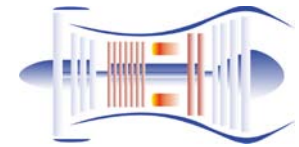
- **Level 1: Users who develop libraries of components.**
 - Model physics knowledge.
 - Model math knowledge.
 - EL code knowledge.



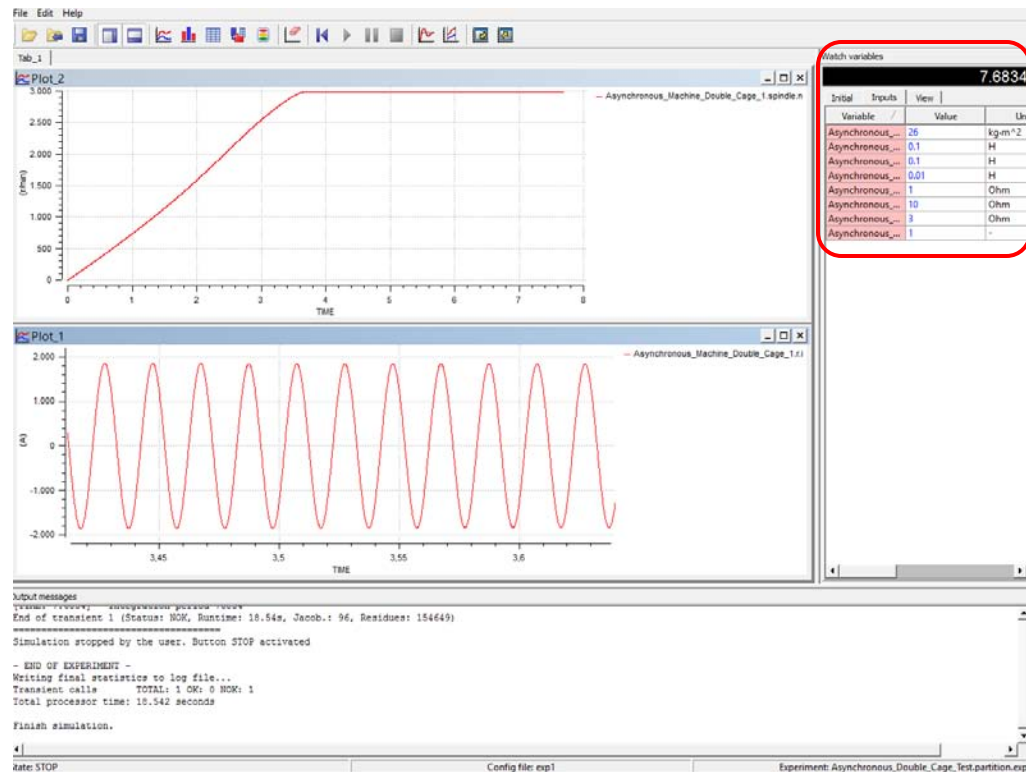


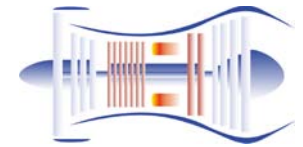
- **Level 2: Users who create models based on existing libraries.**
 - Components deep knowledge.
 - Schematic modelling.



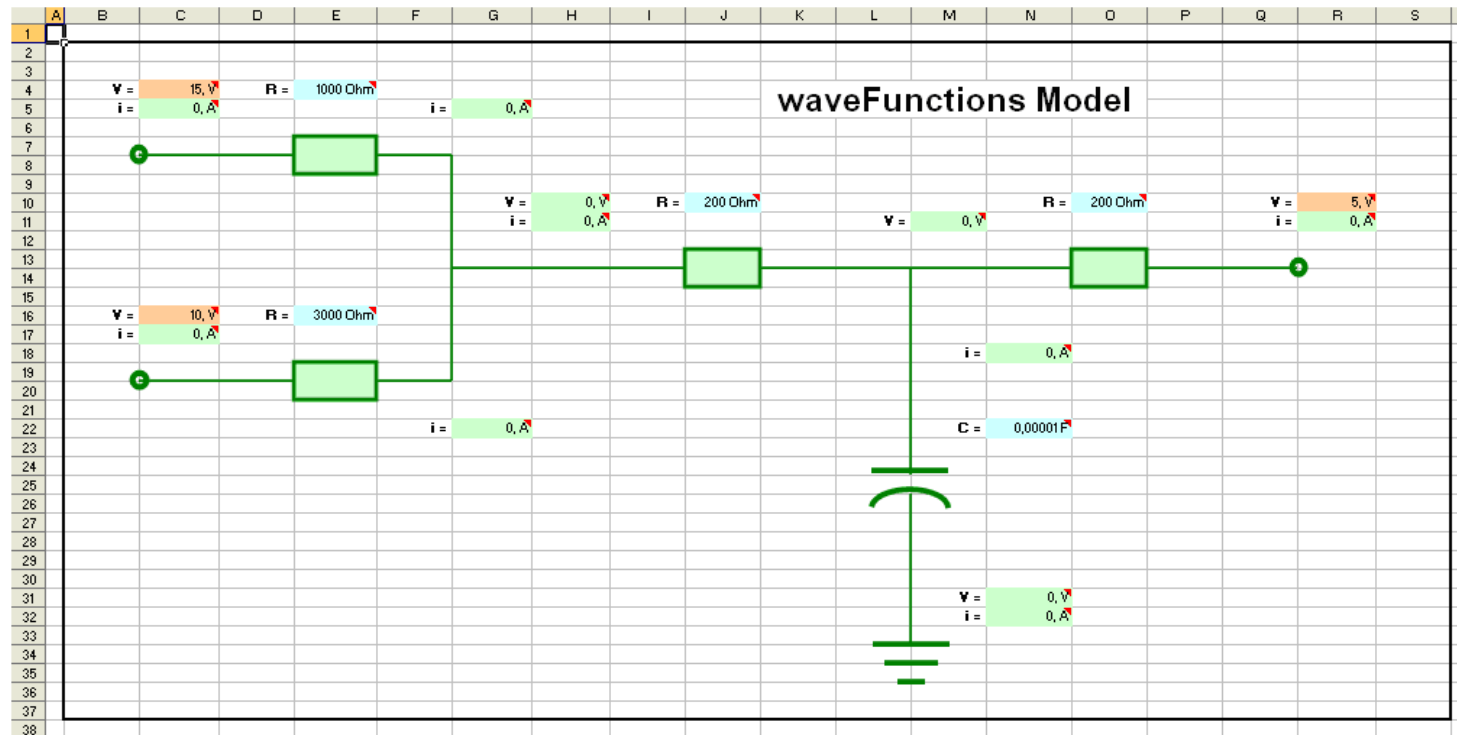


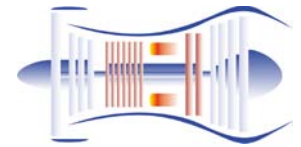
- **Level 3: Users who run simulations from existing models.**
 - Running the simulation changing the input data.
 - Background knowledge of the final application.
 - Mainly by means of the monitor.





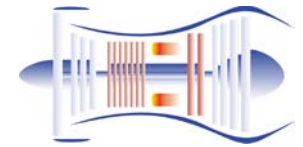
- **Level 4: Users who use exported models.**
 - From EcosimPro to other tools (Excel, Matlab, Simulink) .
 - From EcosimPro as black boxes (Decks).





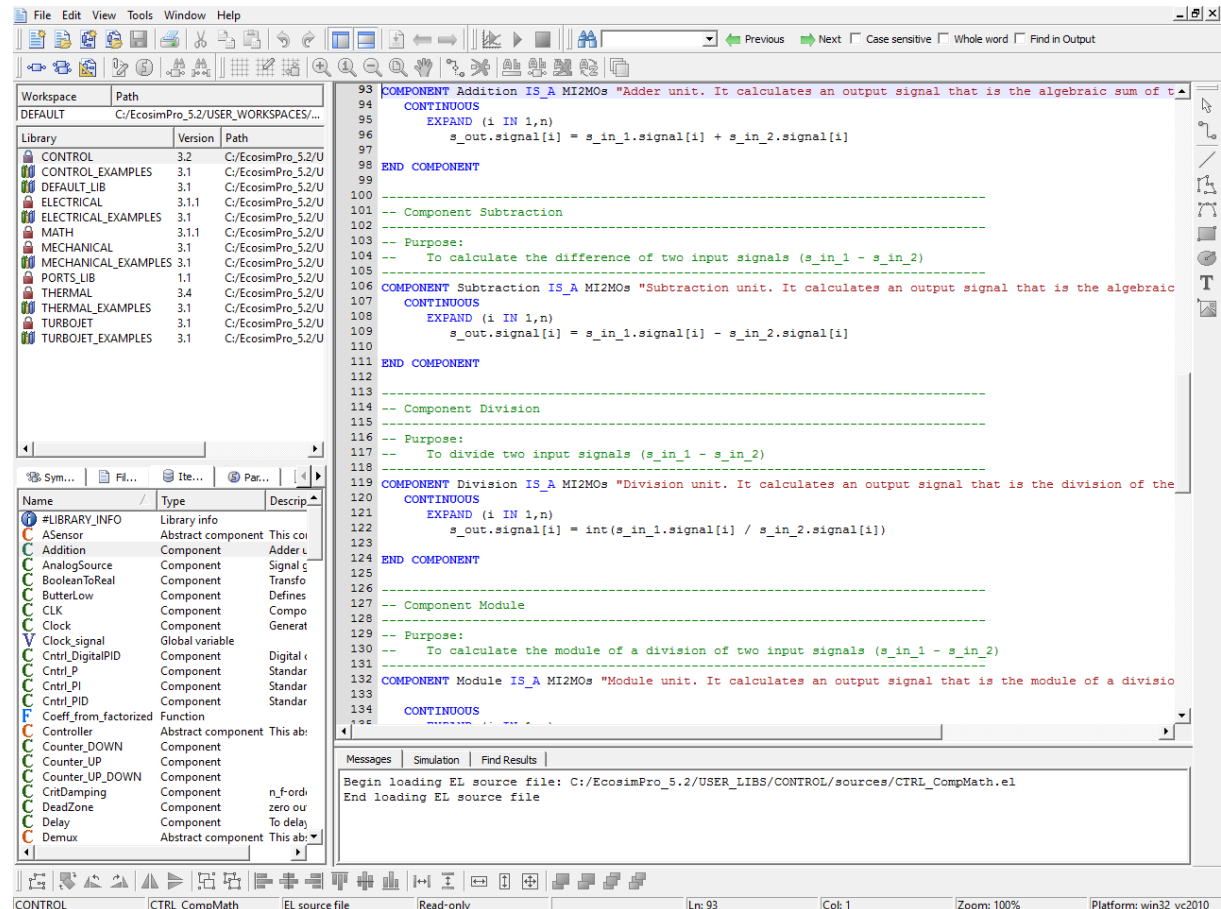
MAIN CONCEPTS

Main Concepts

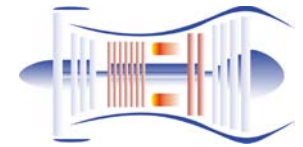


Global View

- Workspace.
- Library.
- Component.
- Schematic.
- Partition.
- Experiment.
- Monitor.
- Deck.

















Main Concepts

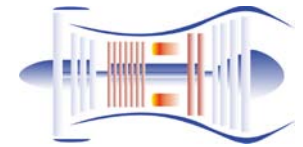


WorkSpace

- Set of libraries related to a specific simulation environment and configuration setup.
- Each user may create different workspaces according to the different developing areas or projects.

Name	Version
 DEFAULT	1.3
 CONTROL	4.0.0
 CONTROL_EXAMPLES	3.1.2
 DEFAULT_LIB	3.1.2
 ELECTRICAL	3.1.2
 ELECTRICAL_EXAMPLES	3.1.2
 MATH	3.1.2
 MECHANICAL	3.1.2
 MECHANICAL_EXAMPLES	3.1.2
 PORTS_LIB	1.1.2
 THERMAL	3.4.2
 THERMAL_EXAMPLES	3.1.2
 TURBOJET	3.1.2
 TURBOJET_EXAMPLES	3.1.2

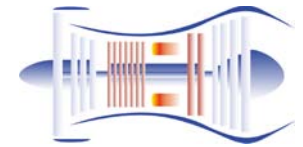
Main Concepts



Library

- Collection of elements related to a specific simulation discipline (Electrical, Mechanical, Thermal, etc).
- Library area: Files, Items, Symbols, Partitions and Experiments related to the library.

Name	Type	Description
#LIBRARY_INFO	Library Info	
Example_CoupledClutches	Component	
p2	Partition	
exp1	Experiment	
Example_Damper	Component	
p1	Partition	
exp1	Experiment	
Example_DriveTrain	Component	
p2	Partition	
exp1	Experiment	
Example_GearFriction1	Component	
p2	Partition	
exp1	Experiment	
Example_GearFriction2	Component	
p2	Partition	
exp1	Experiment	
Example_Spring	Component	
p1	Partition	
exp1	Experiment	
Example_T_Stop	Component	
p1	Partition	
exp1	Experiment	

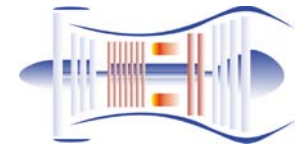


Component

- The Components are the basic elements in EcosimPro.
- They are usually related to a physical element such as compressors, valves, pipes, pumps, ...
- Their formulation describes the behaviour of the related physical element.

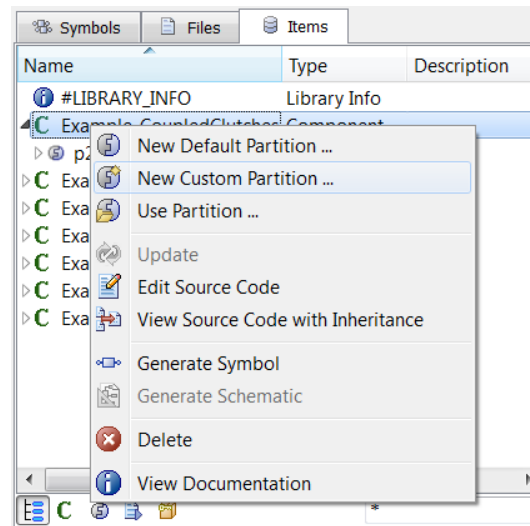
Name	Type	Description	Pa
MIMOs	Abstract Co...	This abstrac...	
MO	Abstract Co...	This abstrac...	
MathFunction	Component	To perform ...	
MathOption	Enumerative		
Maximum	Component	Outputs the ...	
Minimum	Component	Outputs the ...	
Module	Component	Module unit...	
Move	Component	Move stand...	
Mux	Abstract Co...	This abstrac...	
Mux2	Component	Multiplexer ...	

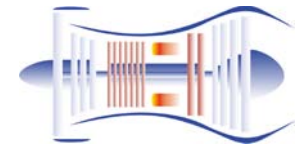
```
29 -----
30 -- Component Minimum
31 -----
32 -- Purpose:
33 --   Outputs the minimum input value.
34 -----
35 COMPONENT Minimum IS_A MI2MOS "Outputs the minimum input value"
36 CONTINUOUS
37   EXPAND (i IN 1, n)
38     s_out.signal[i] = ZONE (s_in_1.signal[i] < s_in_2.signal[i]) s_in_1.signal[i]
39                       OTHERS
40                           s_in_2.signal[i]
41 END COMPONENT
42
```



Partition

- Mathematical model associated to a component.
- Defines the inputs and outputs of the simulation.
- Generates a definite solution for the complete equation system.

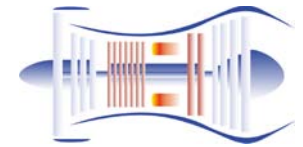




Experiment

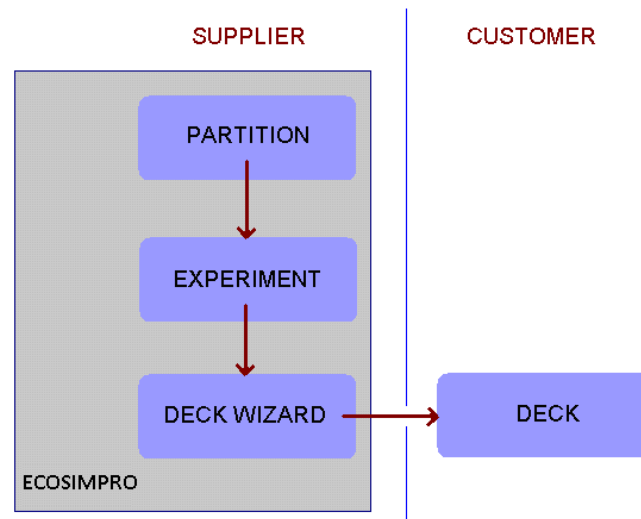
- Configures the simulation on a defined partition of a component.
- Customizes the reports of the simulations
- Generates a sequence of studies of the behaviour of the component

```
EXPERIMENT exp1 ON Component.Partition
DECLS
--
INIT      -- set initial values for variables
--        -- Dynamic variables
--        -- Algebraic variables
BOUNDS    -- set expressions for boundary variables: v = f(t,...)
--
BODY
  REPORT_MODE = IS_STEP
  TIME = 0
  TSTOP = 2
  CINT = 0.005 --0.01
  INTEG()
END EXPERIMENT
```

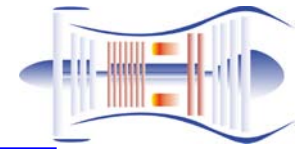


Deck

- Standalone Executable file that encapsulates models and experiments
- From the user point of view, a Deck is like a black – box
- Allows the connection with other software

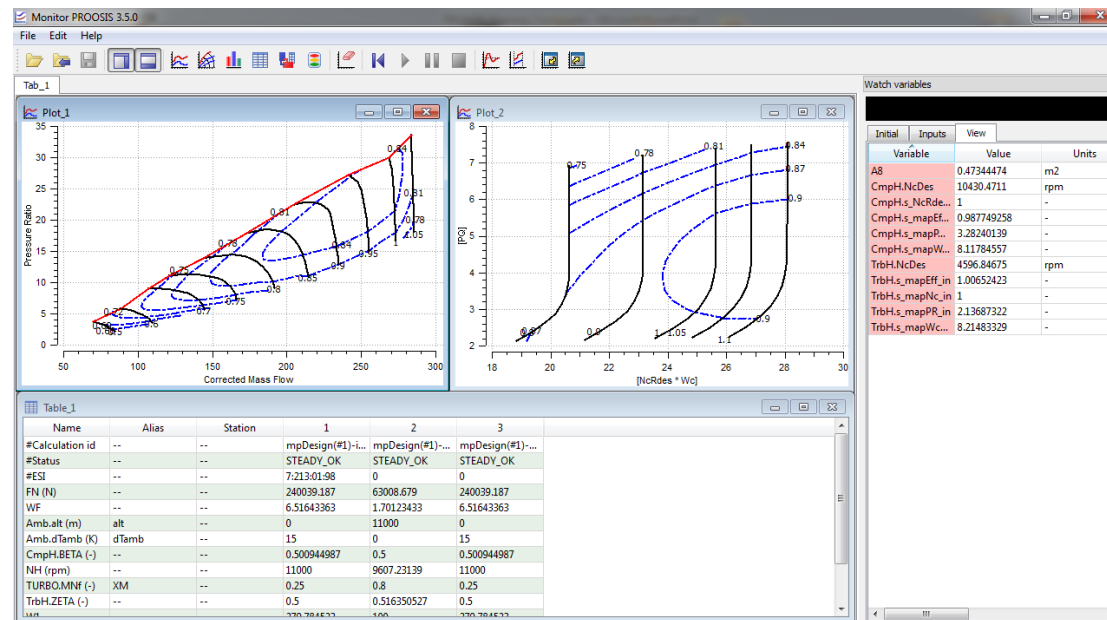


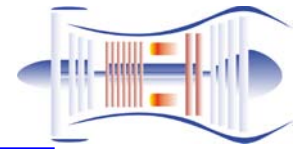
Main Concepts



Monitor

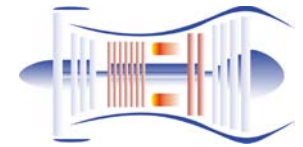
- Graphical Interface to analyze the results of simulations in detail
- Representation of variables and performance maps.





GRAPHICAL USER INTERFACE

Graphical User Interface



General Toolbar

Work Space Libraries

Library Elements

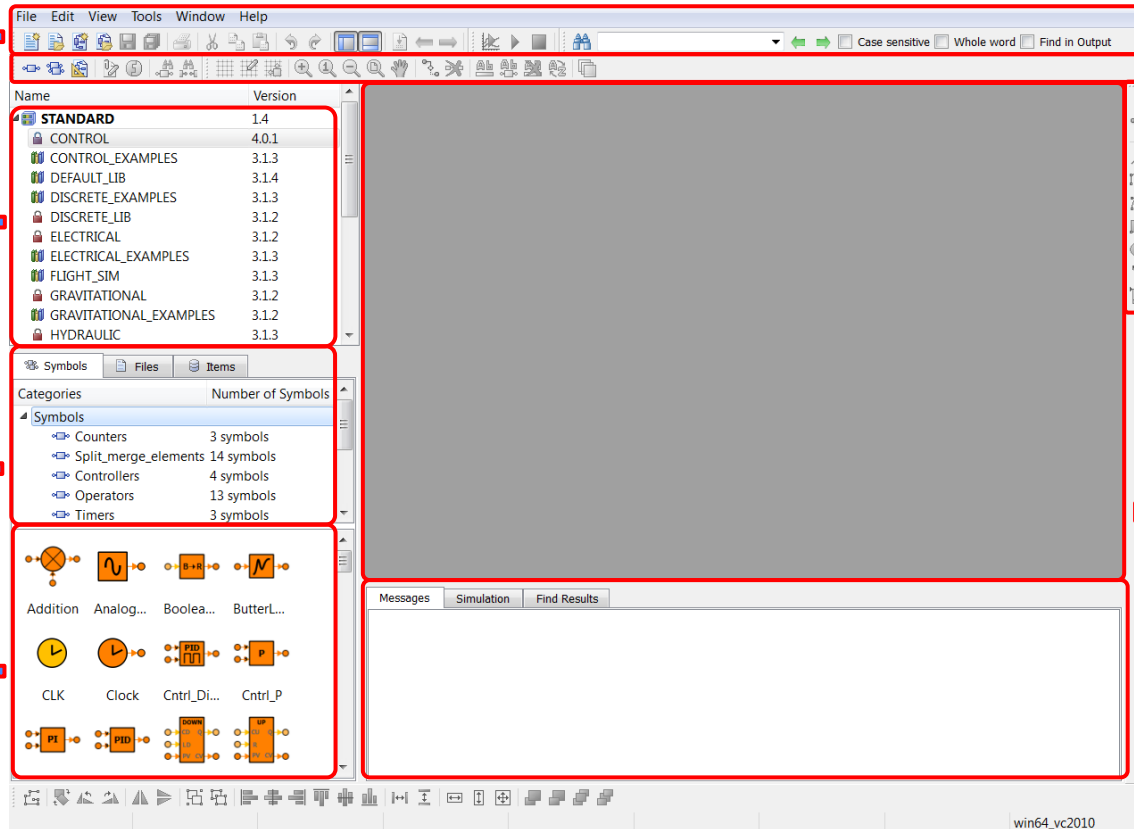
Library Palette

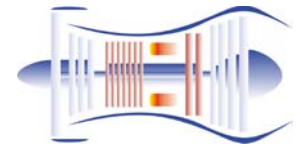
Schematic Toolbar

Graphical Toolbar

Editing Area

Messages Area





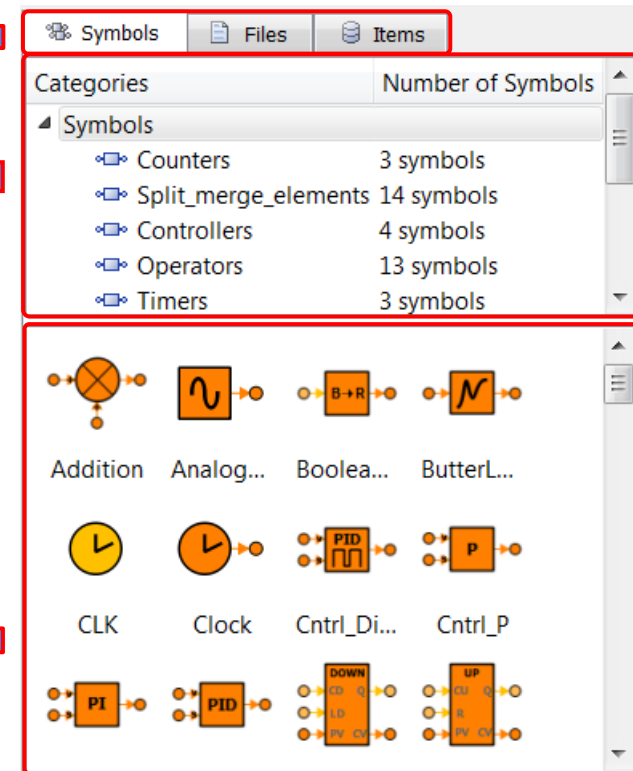
Library View

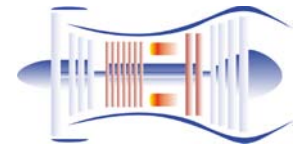
Library Tabs:

- Symbols
- Files
- Items

Palette Subcategories.

Palette: Library Symbols





Library View

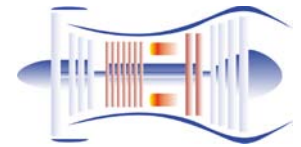
Library Tabs:

- Symbols
- **Files**
- Items

Files:

- Component Code
- Schematics

Symbols Files Items		
Files		Type
CTRL_Common.el		Source code
CTRL_CompASensor.el		Source code
CTRL_CompBistable.el		Source code
CTRL_CompComparison.el		Source code
CTRL_CompContinuous.el		Source code
CTRL_CompControllers.el		Source code
CTRL_CompCounters.el		Source code
CTRL_CompDigitalPID.el		Source code
CTRL_CompDiscrete.el		Source code
CTRL_CompLogical.el		Source code
CTRL_CompMath.el		Source code
CTRL_CompMuxers.el		Source code
CTRL_CompNonLinear.el		Source code
CTRL_CompOthers.el		Source code
CTRL_CompSignalTreatment.el		Source code
CTRL_CompSources.el		Source code
CTRL_CompTimers.el		Source code
CTRL_PrivateFunctions.el		Source code
version.el		Source code



Library View

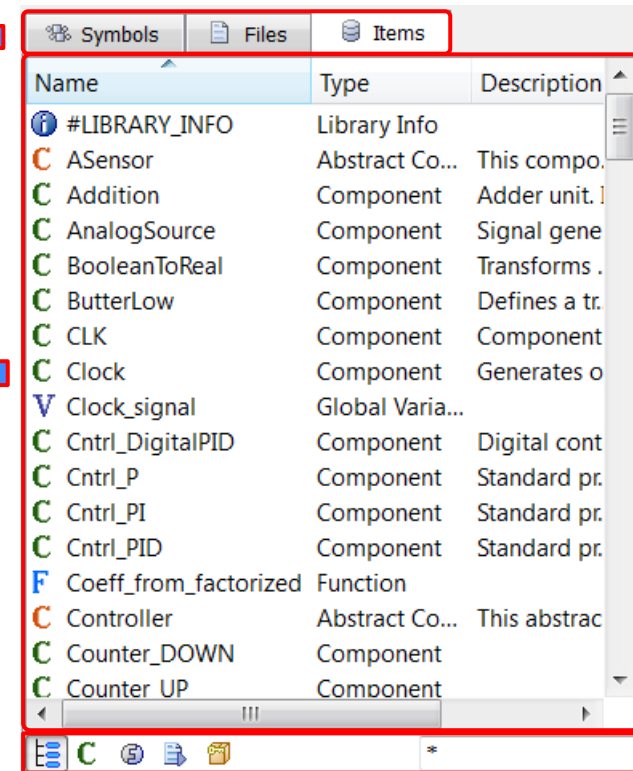
Library Tabs:

- Symbols
- Files
- **Items**

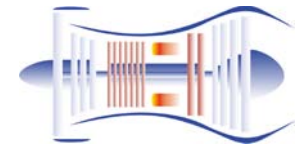
Items:

- Components
- Variables
- Functions
- Partitions
- Experiments

Filters & View



Graphical User Interface



EcosimPro 5.4.0 ENTERPRISE

File Edit View Tools Window Help

Case sensitive Whole word Find in Output

LinearSystem.eds

Name Version Path

- ECOSIMPRO_TRAINING_COURSE 0.0 C:/EcosimP
- CONTROL 4.0.0 C:/EcosimP
- CONTROL_EXAMPLES 3.1.2 C:/EcosimP
- ECOSIMPRO_COURSE_EXAMPLES 0.0 C:/EcosimP
- ELECTRICAL 3.1.2 C:/EcosimP
- MATH 3.1.2 C:/EcosimP
- PORTS_LIB 1.1.2 C:/EcosimP
- THERMAL 3.4.2 C:/EcosimP

Attribute Editor

Library: CONTROL

Type: TransferFunction

Name: process_plant

☒ Show Label

General

Name	Type	Value	Units	Description
PARAMETERS				
n_den	INTEGER	3	-	Order of the denominator ...
n_num	INTEGER	0	-	Order of the numerator pol...
DATA				
output_o	REAL	0	-	Initial value of the output ...
p[0]	ARRAY REAL	{ 1 }	-	Coefficients of numerator ...
q[0]	ARRAY REAL	{ 1,3,2,3,56,0 }	-	Coefficients of denominat...

Documentation ... Default Values Cancel OK

comparator

process_plant

summer_2

Messages Simulation Find Results

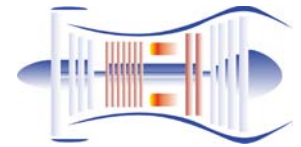
Begin loading Schematic file:
C:/EcosimPro 5.4.0_User_LBS_WS/USER_LBS/CONTROL_EXAMPLES/schematics/+linear+system.eds
End loading Schematic file

CONTROL_EXAMPLES LinearSystem Paper: A4 Landscape Active Layer: Layer_1 Zoom: 76% Pos: (552,597) Platform: win32_vc2010

Attribute Editor

Drag and Drop
Modelling

Graphical User Interface



Simulation Toolbar

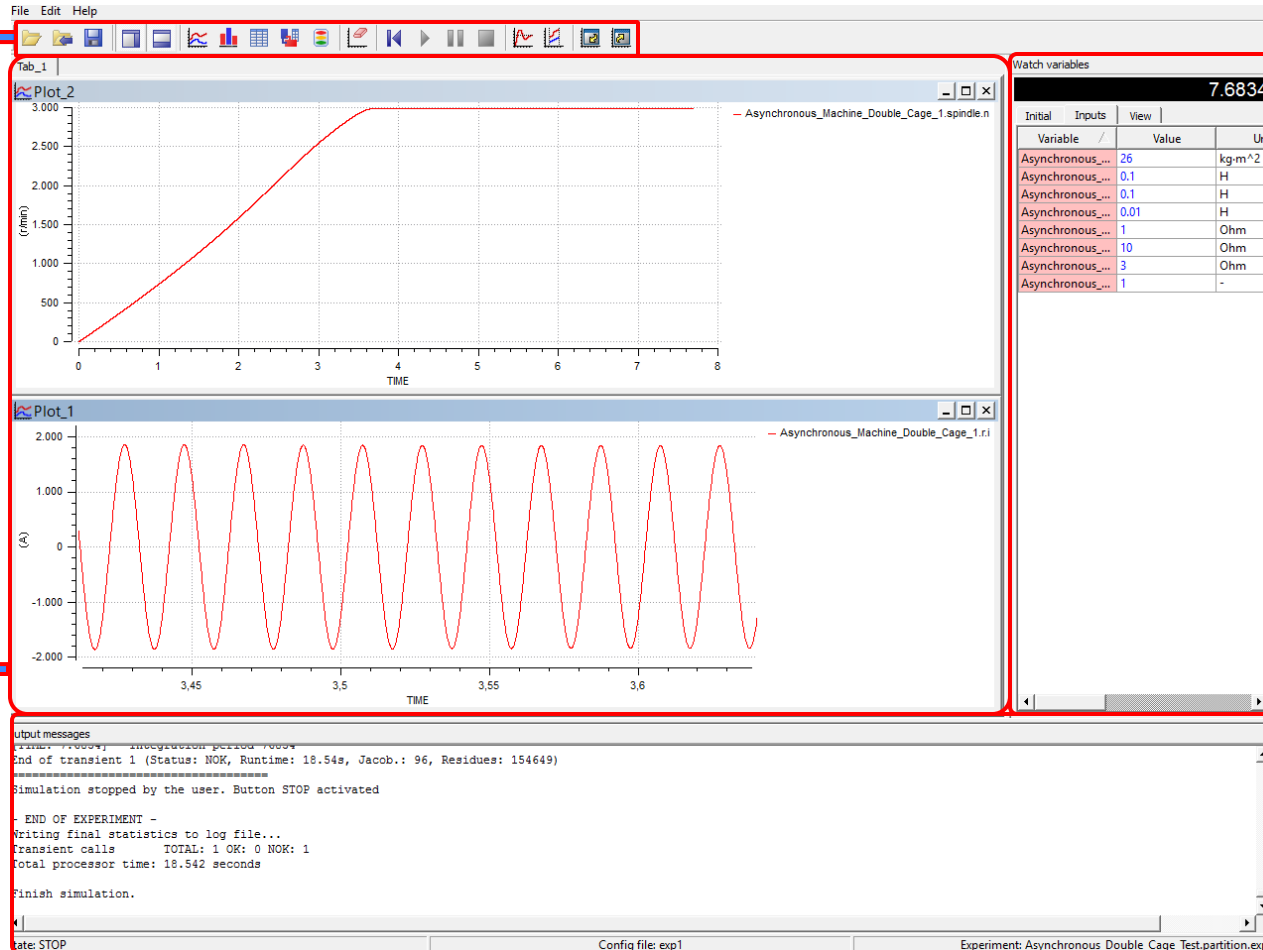
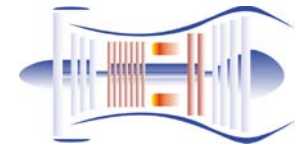
Edit Area

Items Area

The screenshot shows the EcosimPro 5.4.0 ENTERPRISE software interface. The top menu bar includes File, Edit, View, Tools, Window, and Help. Below the menu bar is a toolbar with various icons for file operations, editing, and simulation. The left sidebar, labeled 'Items Area', displays a tree view of the project structure, including libraries like ECOSIMPRO_TRAINING_COURSE, CONTROL, and MATH. The central workspace, labeled 'Edit Area', shows the source code of a file named MATH_Common.el. The code includes comments, constants, and mathematical definitions. The bottom status bar displays the current file (MATH_Common), the source file path, and the zoom level (100%).

```
1  -----
2  -- EA Internacional 2004   EcosimPro 3.3 Simulation Source Code
3  --
4  --   FILE NAME:  MATH_Common.el
5  --   FILE TYPE:  Common elements of the MATH library
6  --   DESCRIPTION: Defines common constants for the MATH library
7  --   NOTES:      Based on EcosimPro MATH library
8  --   AUTHOR:     Ramon Perez Vara
9  --   CREATION DATE: 12-Jan-2004
10 -----
11
12 -- Constants
13 CONST REAL ZERO = 0.                                "Zero value"
14
15 CONST REAL Eps = 1.e-8                                " "
16 CONST REAL Inf = 1.e38                                " "
17
18 CONST REAL Small = 1.e-16                             " "
19
20 CONST INTEGER Inf_int = 2147483647                     " "
21
22 -----
23 -- Mathematical and scientific constants
24 -----
25 CONST REAL E = 2.7182818284590452353602874713527      "Natural logarithmic base"
26 CONST REAL GAMMA = 0.5772156649015328606065120900824  "Euler's Constant"
27 CONST REAL PI = 3.1415926535897932384626433832795     "PI, Archimedes' number"
28
29
30 -----
31 -- Units conversion constants
32 -----
33
34 CONST REAL DtoR = PI / 180.                            "Degree to radian"
35 CONST REAL RtoD = 180. / PI                            "Radian to degrees"
36
```

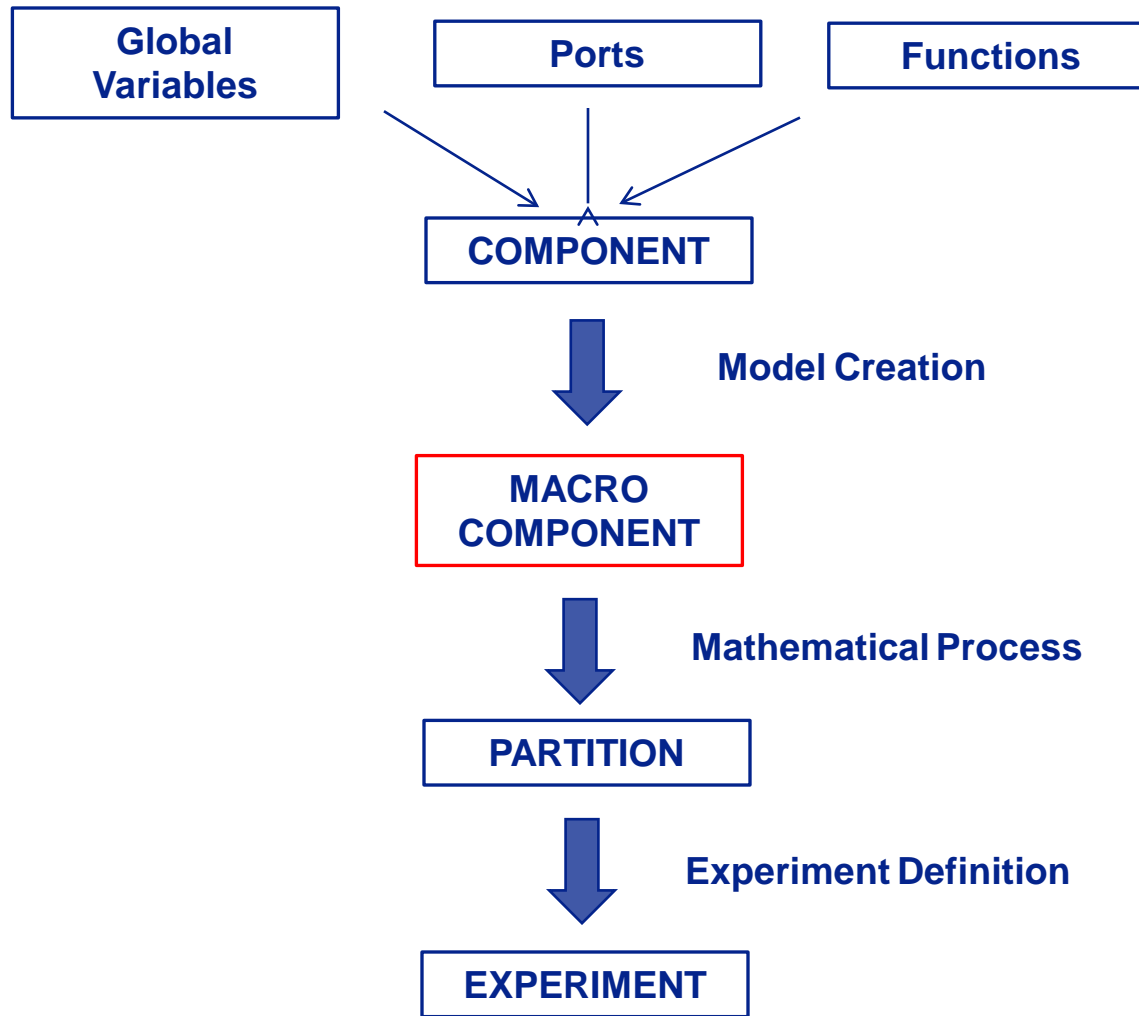
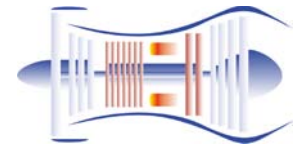
Graphical User Interface



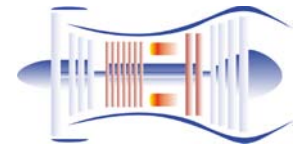
Initial, Inputs
and View Tabs

Messages Area

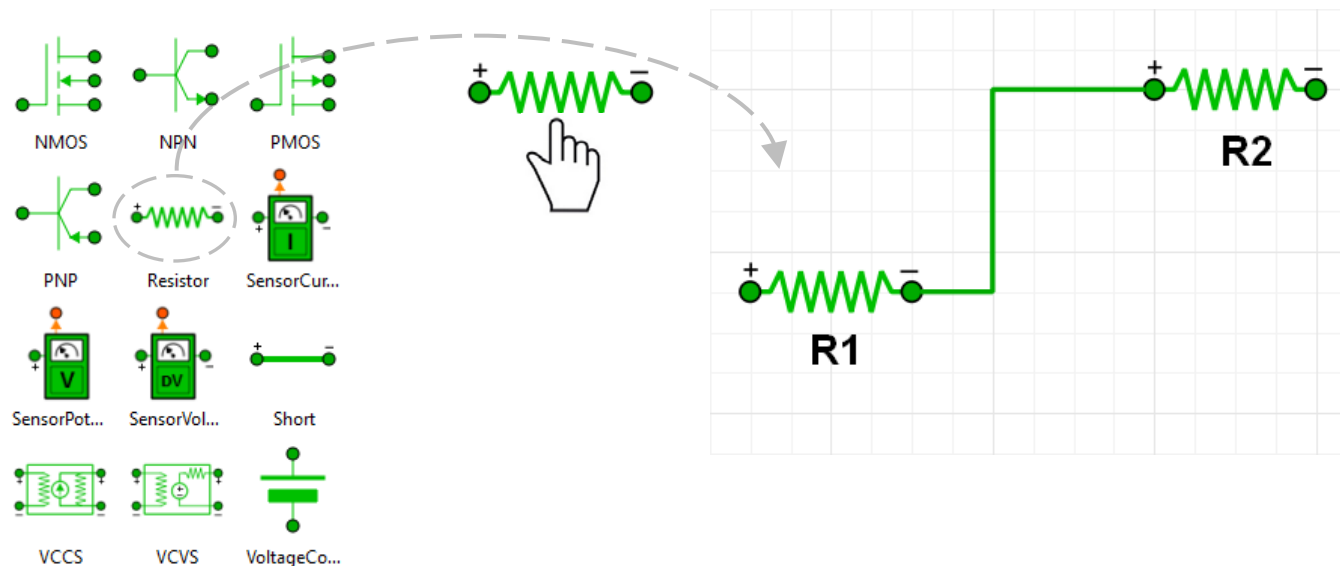
Schematics



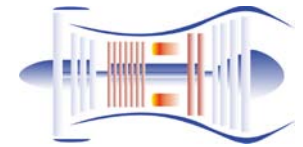
Schematics



- Macro-components can be created graphically by means of Schematic diagrams
- Components from different libraries can be instantiated by **drag & drop**
- **Ports** and **Connections** define the interface between components



Schematics



- When Schematics are compiled, the component is added to the library
- This new item contains the complete formulation of the instances, and their port variables relationships

```
COMPONENT Resistor IS_A OnePort
  "Resistor"
  DATA
    REAL R=1 UNITS u_Ohm "Resistance"
  CONTINUOUS
    R*i = v
END COMPONENT

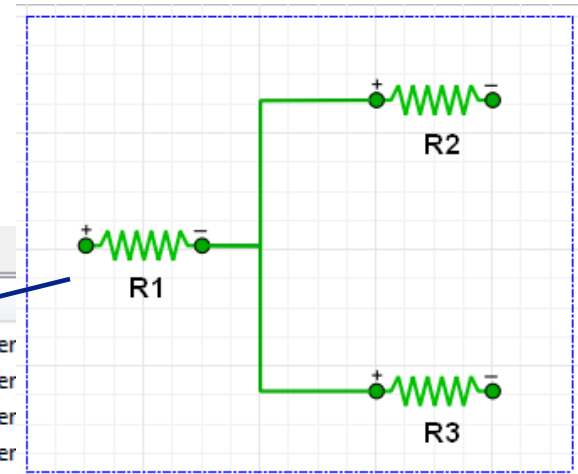
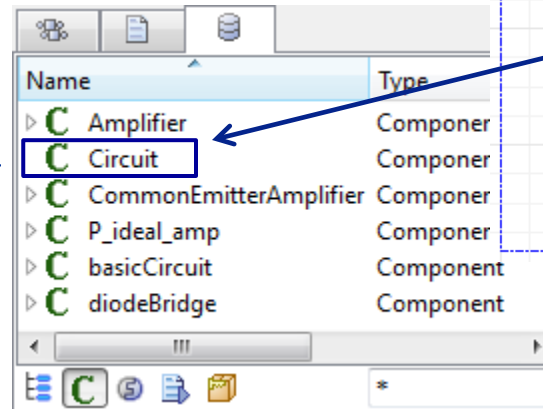
COMPONENT Circuit
  TOPOLOGY

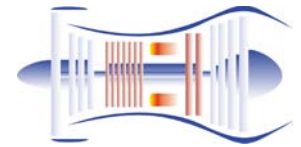
  -----
  -- Coded in Circuit
  -----

  ELECTRICAL.Resistor R1 (R = 1)
  ELECTRICAL.Resistor R2 (R = 1)
  ELECTRICAL.Resistor R3 (R = 1)

  -----
  -- Coded in Circuit
  -----

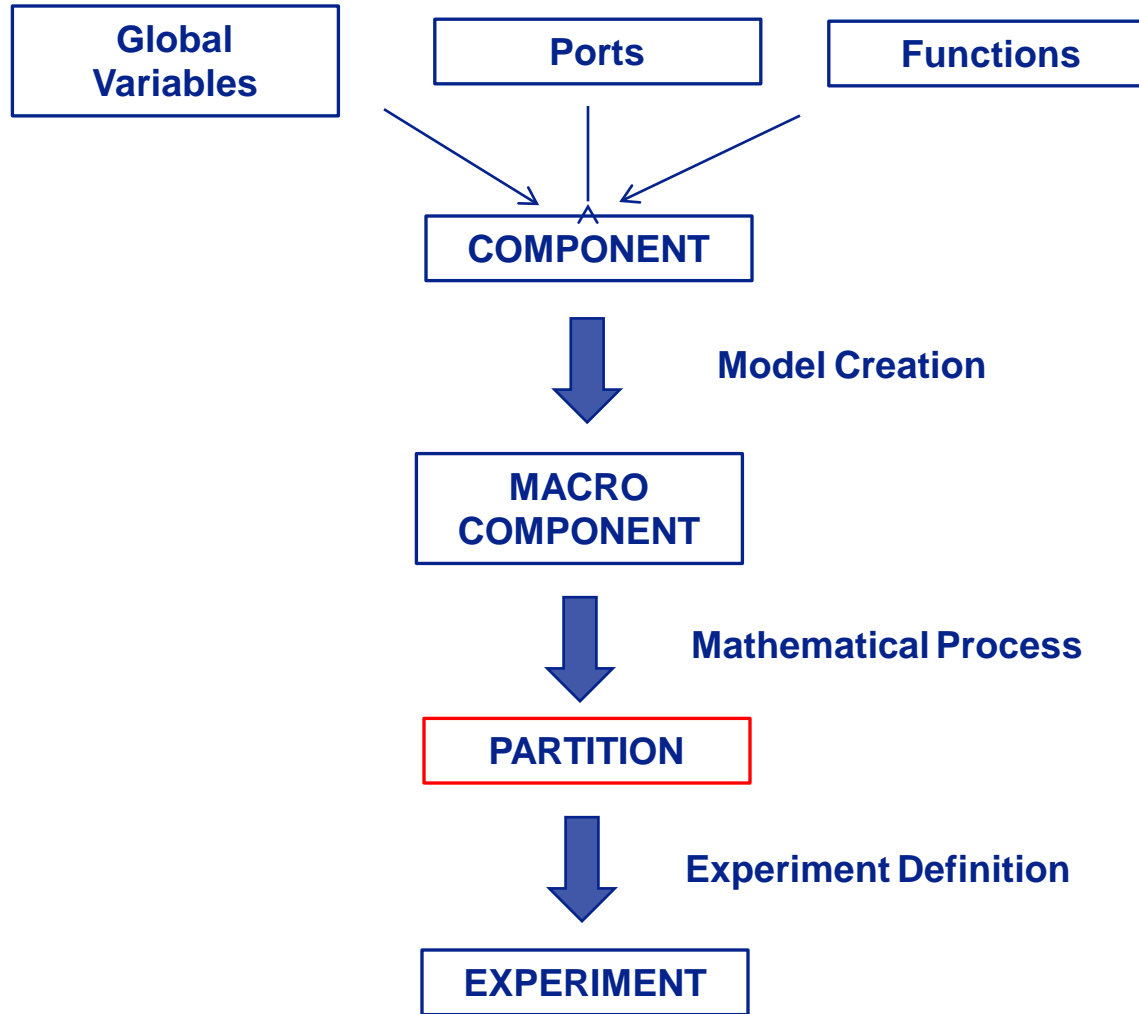
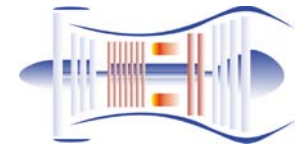
  CONNECT R1.e_n TO R2.e_p
  CONNECT R1.e_n TO R3.e_p
END COMPONENT
```

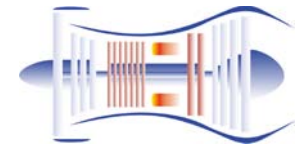




PARTITION

Partitions





System of Equations

- When a Component source code is compiled, all its variables and equations are encapsulated to be reused in the future
- This component contains the equations that must be satisfied during the simulation

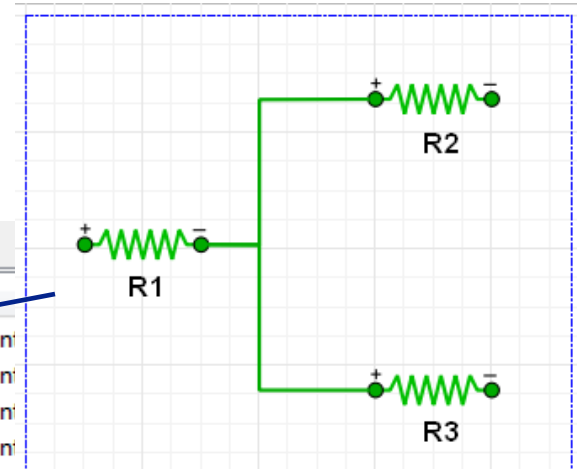
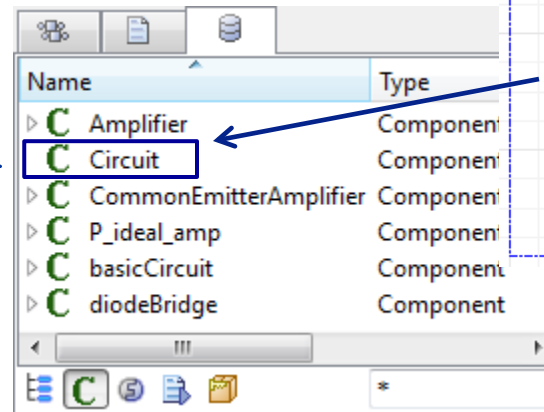
```
COMPONENT Resistor IS_A OnePort
  "Resistor"
  DATA
    REAL R=1 UNITS u_Ohm "Resistance"
  CONTINUOUS
    R*i = v
END COMPONENT

COMPONENT Circuit
  TOPOLOGY

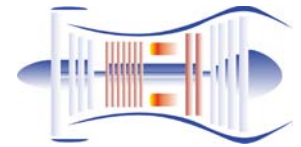
  -----
  -- Coded in Circuit
  -----
  ELECTRICAL.Resistor R1 (R = 1)
  ELECTRICAL.Resistor R2 (R = 1)
  ELECTRICAL.Resistor R3 (R = 1)

  -----
  -- Coded in Circuit
  -----

  CONNECT R1.e_n TO R2.e_p
  CONNECT R1.e_n TO R3.e_p
END COMPONENT
```



Partitions



System of Equations

- These equations are not organized. The system cannot be solved
- The partition organize all the equations and check the number of **boundaries** that must be set by the user:

$$N_{\text{bounds}} = N_{\text{variables}} - N_{\text{equations}}$$

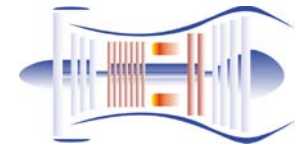
- **Dynamic** variables and **Algebraic** loops are detected

Then, the **Mathematical Model** is ready to be solved if the initial values and Boundaries are given

Name	Type
▶ C Amplifier	Component
▶ C Circuit	Component
partition	Partition
▶ C CommonEmitterAmplifier	Component
▶ C P_ideal_amp	Component
▶ C basicCircuit	Component
▶ C diodeBridge	Component

Sort of **Equations**
Boundaries information
Variables **Initialization**

Partitions



System of Equations

- The **Mathematical Model** of any partition can be analyzed to check the sequential solving process

The screenshot shows the PROOSIS software interface. On the left, a tree view lists components: #LIBRARY_INFO (Library Info), Circuit (Component), and partition (Partition). The 'partition' component is selected. A context menu is open over the 'partition' component, showing options: Simulate in Monitor, New, Edit ..., Validate, Paste Group, Compile Experiments, Validate Wizard Experiments, View Mathematical Model (highlighted), View External Objects ..., Delete, Export Files ..., and Open Containing Folder.



GENERAL STATISTICS

INFO	#
Number of equations:	10
Number of boxes (coupled subsystems of equations):	0
Number of linear boxes:	0
Number of nonlinear boxes:	0
Number of input DATA:	3
Number of input BOUNDARY:	3
Number of output EXPLICIT :	10
Number of output DYNAMIC/DERIVATIVE:	0
Number of output ALGEBRAIC:	0
Size of Jacobian matrix (DYNAMIC+ALGEBRAIC):	0x0
Default integration method:	DASSL

TYPE OF VARIABLES

TYPE	VARIABLE	DATA	CONSTANT
REAL	13	3	0
INTEGER	0	0	0
STRING	0	0	3
TABLE	0	0	0

GLOBAL FLAGS:

FLAG	VALUE
Remove derivatives	FALSE
Inhibit automatic reduction of equations	FALSE
Remove not used variables	FALSE
Generate code to check mathematical functions	TRUE
Generate code to analyse the performance of functions	FALSE
Obfuscate partition and experiment C++ generated code	FALSE

BOUNDARIES:

#	NAME	ALIAS	UNITS	DESCRIPTION	INITIAL
1	R1.e.p.v	v1	V	Potential at pin	
2	R2.e.p.i	i2	A	Current flowing into the pin	
3	R3.e.p.i	i3	A	Current flowing into the pin	

JACOBIAN INDEPENDENT VARIABLES:

POS	VARIABLE	ALIAS	CATEGORY	UNITS	DESCRIPTION	INITIAL	RESIDUE	EQUATION
-----	----------	-------	----------	-------	-------------	---------	---------	----------

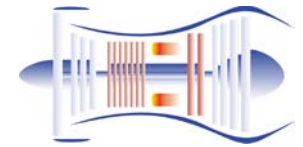
SORTED EQUATIONS:

```
##eqts 10
[E-1] i1 = i2 + i3
[E-2] R1.v = R1.R * i1
[E-3] R3.v = R3.R * i3
[E-4] v3 = v1 - R3.v - R1.v
[E-5] vc = v3 + R3.v
[E-6] R1.e_n.i = -i1
[E-7] R2.e_n.i = -i2
[E-8] R2.v = R2.R * i2
[E-9] v2 = v3 + R3.v - R2.v
[E-10] R3.e_n.i = -i3
```

SORTED EQUATIONS:

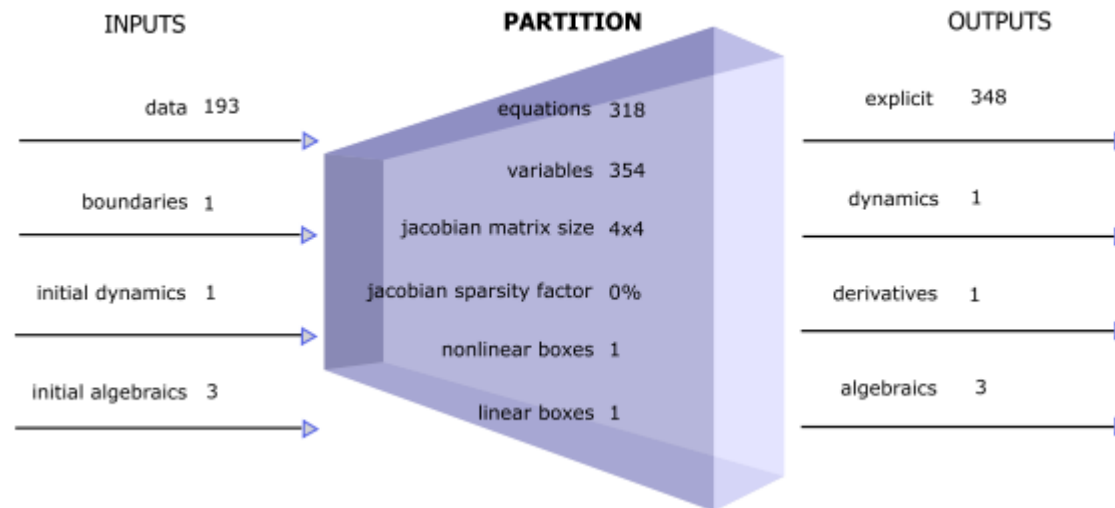
```

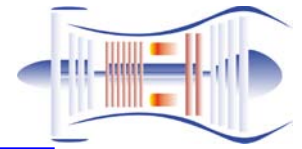
##eqts 10
[E-1] i1 = i2 + i3
[E-2] R1.v = R1.R * i1
[E-3] R3.v = R3.R * i3
[E-4] v3 = v1 - R3.v - R1.v
[E-5] vc = v3 + R3.v
[E-6] R1.e_n.i = i1
[E-7] R2.e_n.i = i2
[E-8] R2.v = R2.R * i2
[E-9] v2 = v3 + R3.v - R2.v
[E-10] R3.e_n.i = i3
    
```



Component vs. Partition

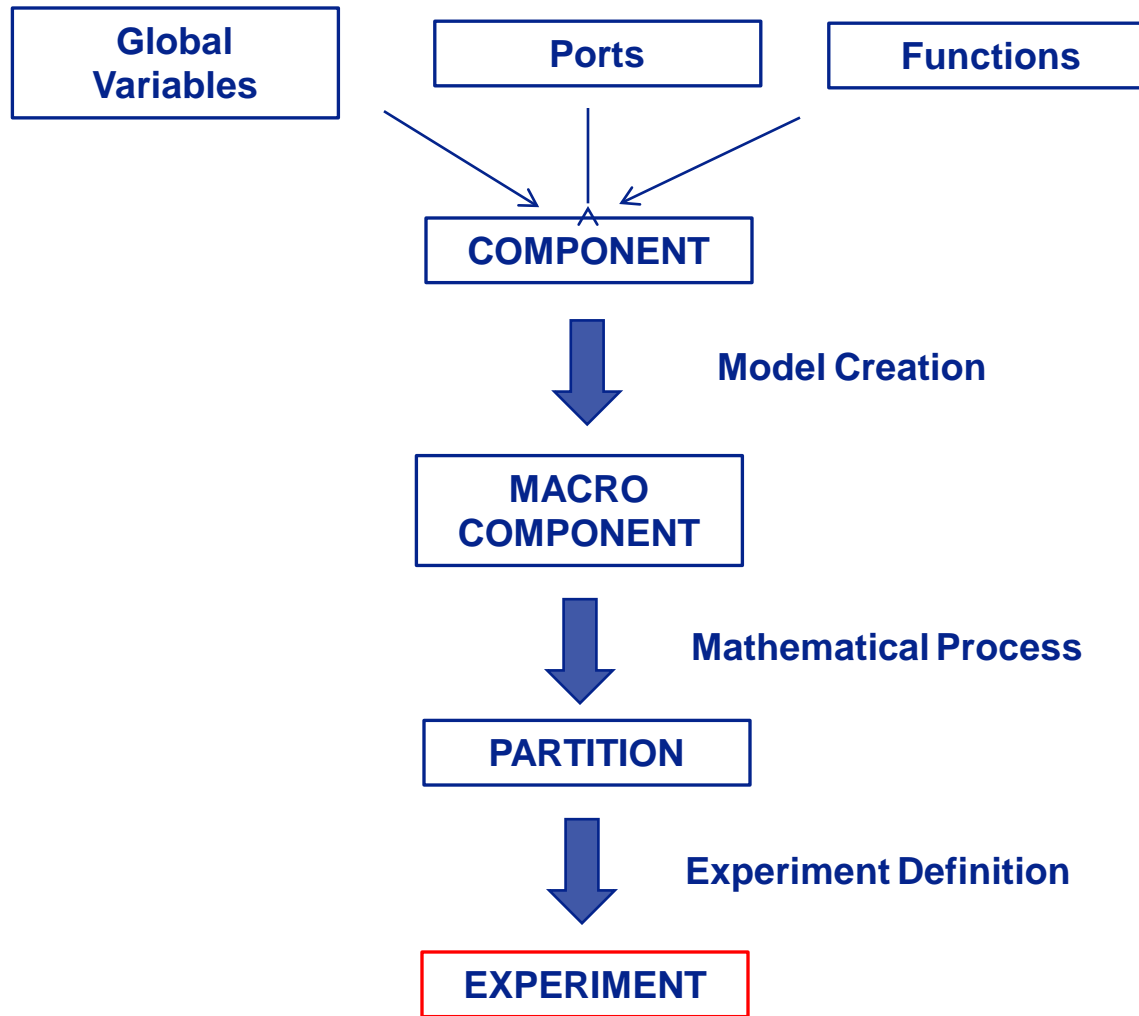
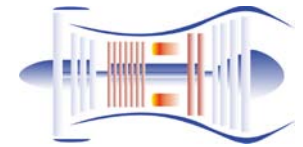
- The **Component** provides the set of equations to be solved. The **Partition** organizes them to obtain a consistent system of equations
- The Partition defines the **INPUTS** and **OUTPUTS** of the model.

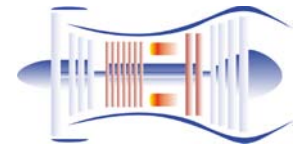




EXPERIMENT

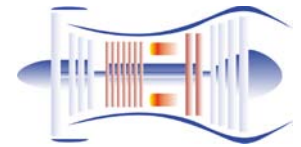
Experiments





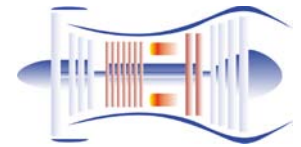
Simulation Definition

- The Partition defines the **Inputs** and **Outputs**. Different calculations can be defined for it in a Experiment
- The Experiment is a sequence of statements which allows the user to:
 - **Initialize data**
 - **Solve** the **system of equations** defined in the partition
 - Create **extra functions** to manage the model variables
 - Define **extra equations** to extend the model equations
 - Prepare the model for **external connections**



Simulation Definition

- Typical calculations performed in experiments:
 - **Steady**
 - **Transient**
 - Design
 - Parametric
 - Sensitivity Analysis
- The Experiment can be used to define complex sets of calculations to perform complete analysis of models
- The **Experiment Wizard** provides an intuitive interface to create automatically the experiment code



Experiment Wizard

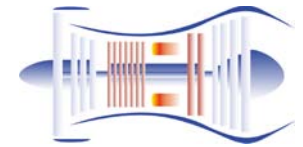
Create experiment code requires a **deep knowledge** of PROOSIS functions

The Experiment Wizard helps the user to generate experiments **graphically**

Wizard experiments are **easier to maintain** when models/partitions change

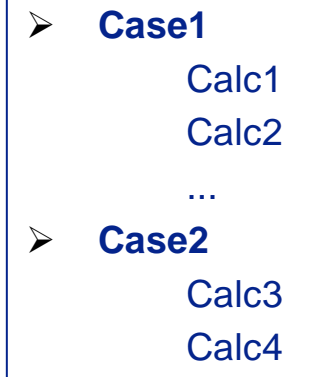
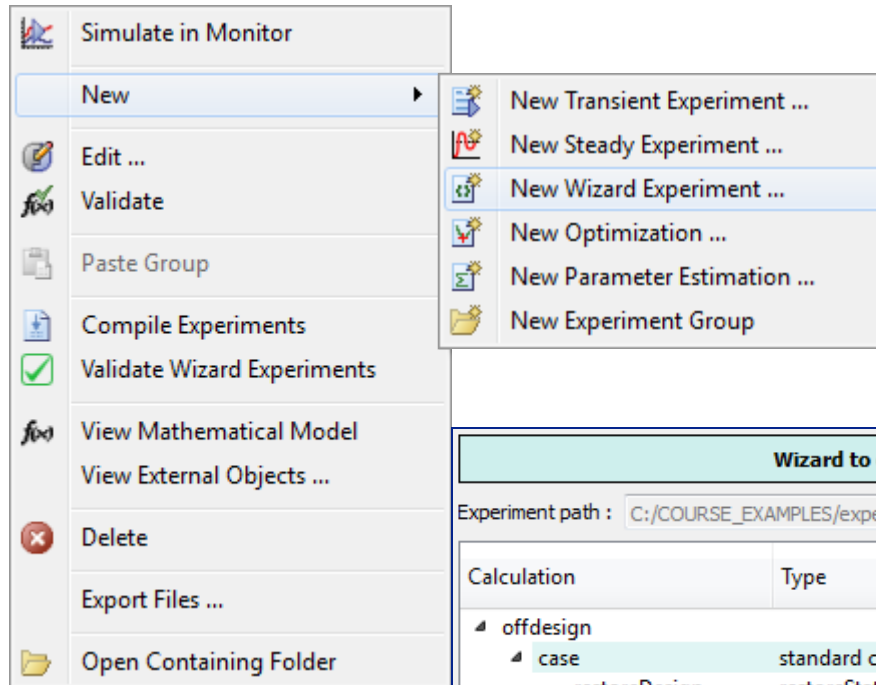
Complex experiments can be generated

Experiments

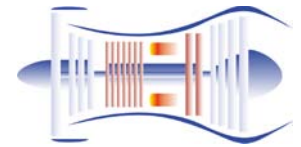


Experiment Wizard

The Wizard experiments have the following structure:



Wizard to create and edit an experiment with multiple calculations.					
Experiment path : C:/COURSE_EXAMPLES/experiments/+turbo_+t+j_+o+d.partition/offdesign/offdesign.cal.xml					
Calculation	Type	Description	Init algebraic and dynamic with previous calculation	Run calculation	Do not plot
offdesign					
case	standard case	Description	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
restoreDesign	restoreState	Restore State	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
flightConditions	data	Data	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
OL	parametric	Parametric	<input type="checkbox"/>	<input checked="" type="checkbox"/>	



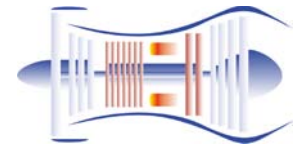
Experiment Wizard

CASE: Group of calculations

- **Standard:** Executes the calculations sequentially
- **Parametric:** Create a loop to execute a set of calculations iteratively.

CALCULATION: Series of statements to set the Inputs/Outputs and solve the Mathematical Model.

- **Simulations:** transient, steady, sensitivity, parametric, ...
- **Functions:** function call, data, data cardpack
- **Reset:** reinitialize variables, execute init blocks, ...
- **Report management:** save state, restore state

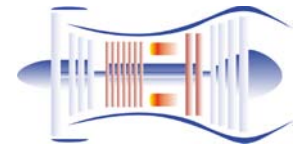


Experiment Wizard

TYPE OF CALCULATIONS

- **Steady:** Calculate steady state point. Dynamics are calculated iteratively (as Algebraics)
- **Extended Steady:** Calculate an steady point. Design data are calculated iteratively to fulfil extra closure equations
- **Transient:** Simulate time evolutions. Dynamics variables are initialized by the user
- **Extended Transient:** Transient simulation. Boundaries are calculated to fulfil extra closure equations
- **Parametric:** Steady / Transien loops
- **Sensitivity:** Calculate input (Data or Bounds) sensitivity in Steady calculation results

Experiments



DATA

$C1 = 1e-3 \text{ F}$

$L1 = 10 \text{ H}$

$R1 = 2e3 \text{ Ohm}$

$V = 10 \text{ V}$

BOUNDARY

$V3 = 100 \text{ V}$

DYNAMIC

$C1.v = 0 \text{ V}$

$i2 = 0 \text{ A}$

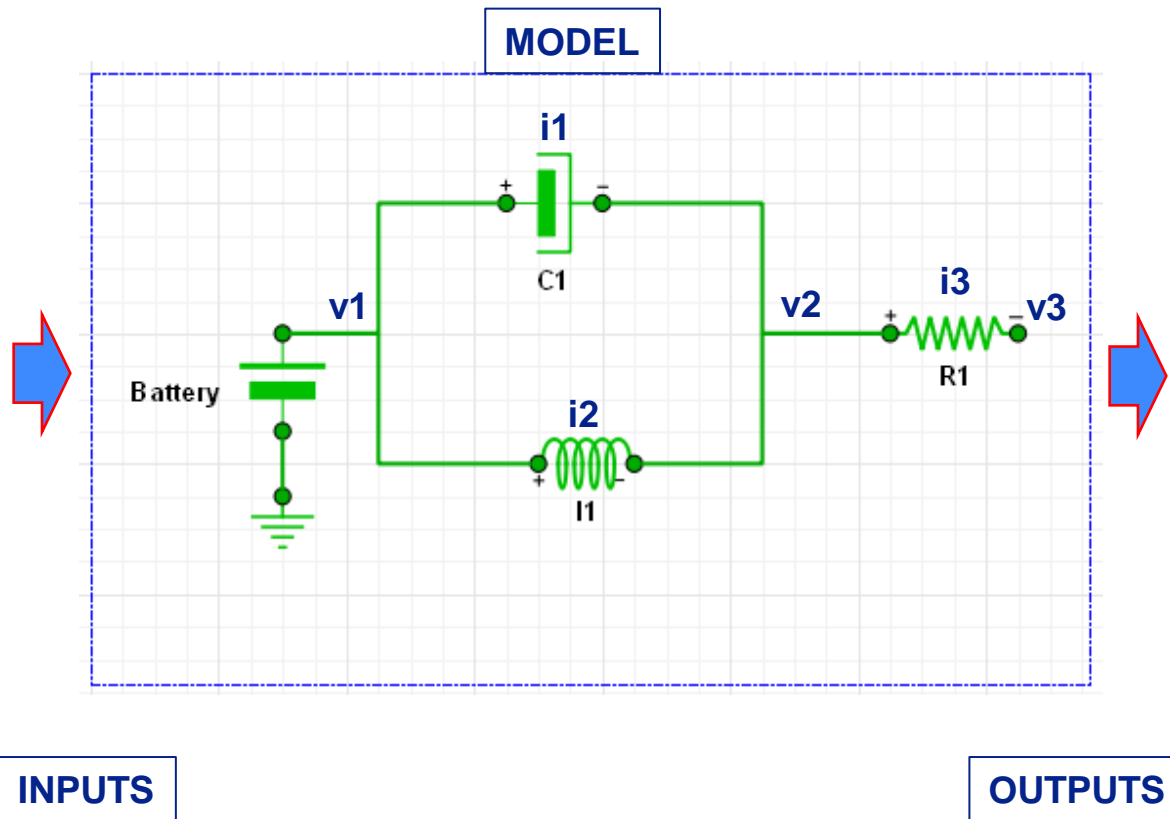
CONFIG

$CINT = 1E-3$

$TIME = 0$

$TSTOP = 10$

Electrical Example



EXPLICIT

$v1, v2$

$i3$

...

DYNAMIC

$C1.v$

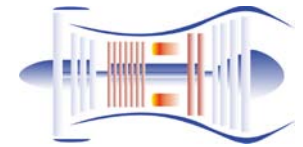
$i2$

DERIVATIVE

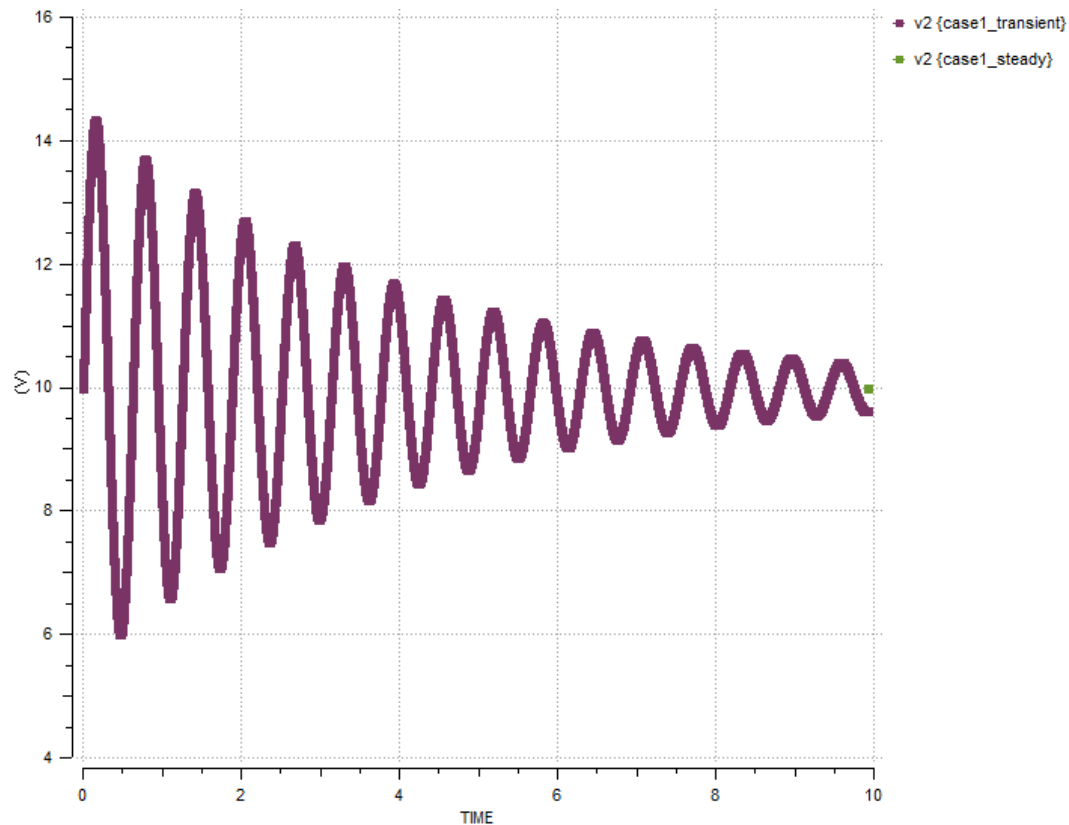
$C1.v$

$i2'$

Experiments

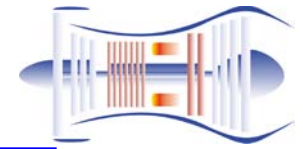


Transient vs. Steady

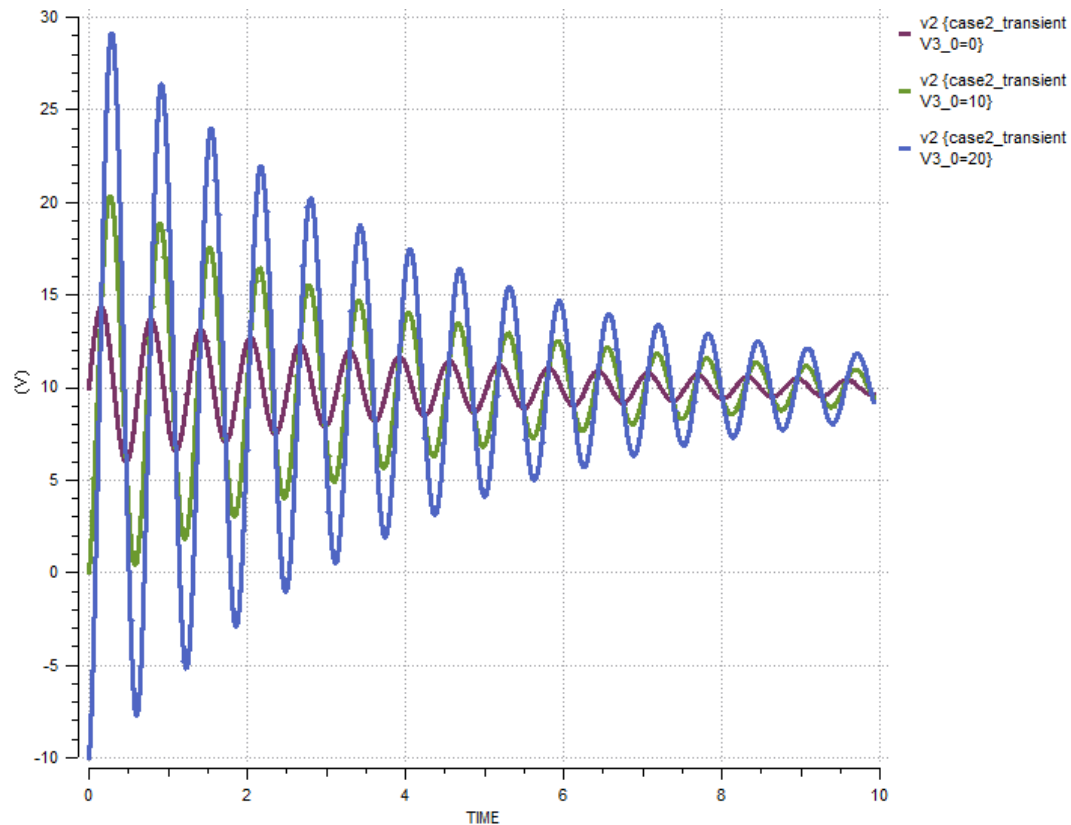


TRANSIENT: the violet line shows the evolution of the **v2** variable from $t=0$ s to $t=10$ s

STEADY: the green point represents the value of **v2** when the system becomes stable and does not change



Parametric Study



PARAMETRIC STUDY:

Loops of Transient or Steady calculations carried out to analyze the influence of:

- Initial states
- Component characteristics
- Boundary conditions
- ...