



Be challenging, be smart: BE CAE & Test!



A COMSOL App to analyze bacteria lethality during sterilization processes



<http://www.be-caetest.it/>

Florence, 22-24/10/2024



About us...

BE CAE & Test provides **consultancy services** to industrial partners by exploiting **CAE software** and developing **experimental testing**.



<http://www.be-caetest.it/>

[BE CAE & Test](#)

info@be-caetest.it

+39 095 216 6426

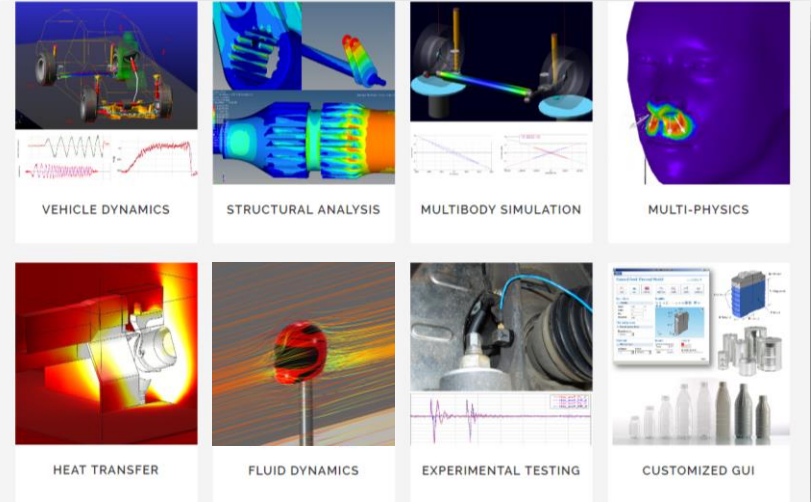


ITALY

- Viale Africa, 170 - 95129 Catania (CT)
- Via Toscana, 104 - 41053 Maranello (MO)

SPAIN

- Calle Impresores, 20 - 28660 Boadilla del Monte (Madrid)



COMSOL Certified Consultants



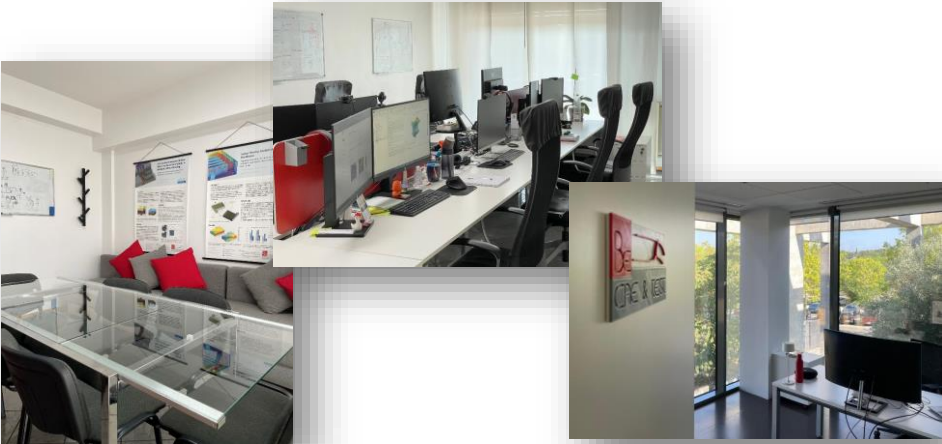
Italy

BE CAE & Test ←

Everywave Srl

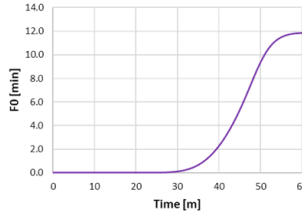
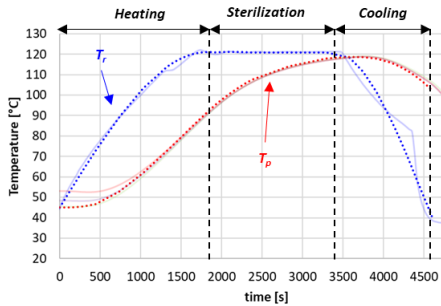
Polibrixia srl

<https://www.comsol.it/certified-consultants>



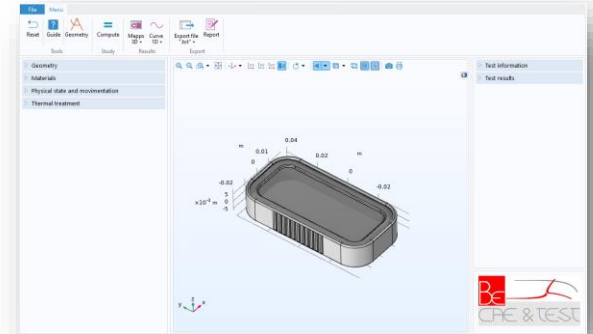
Introduction

- About food safety
- Theoretical notes



Method and validation

- Food safety parameters analysis
- Implementation: model nodes
- Num-exp validation
- Result



API construction

- Forms interface.
- How to choose geometry, materials and thermal cycle
- Results.

Remarks and conclusion

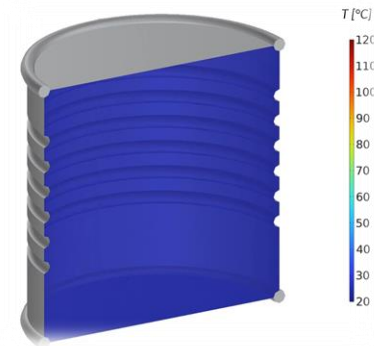


Food safety

- Historically, food safety is a very **critical aspect** in the field of the **food industry**. Making a product safe means **preserving the quality** for a certain period, so that it can be considered non-harmful for consumers and **be marketed**. At date, there are **specific protocols** and treatment useful to guarantee food quality.



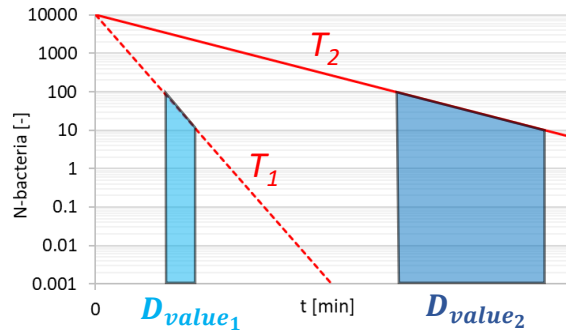
- Degradation** of canned foods depends on the activity and **number of bacteria** inside.



- Our work with **COMSOL** was to create a **stand-alone application** that could **predict** bacteria reduction and **product safety** in **canned food**, related to specific treatments.

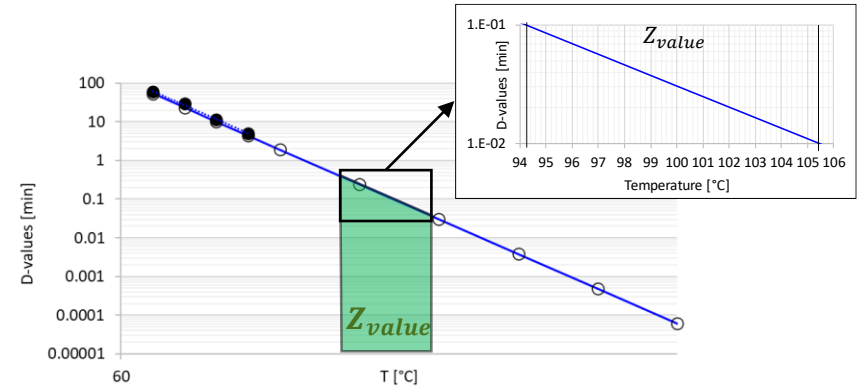
Bacteria heat sensitivity parameters

Bacteria are sensitive to **high temperatures**. During heat treatment, they **start reducing** and can be completely **neutralized** under specific **temperatures** and **time**.



Red lines = Number of surviving bacteria (N) as a function of the time (t) when a specific constant temperature (T_i) are applied.

D_{value} = time required to reduce the number of bacteria by a logarithmic order



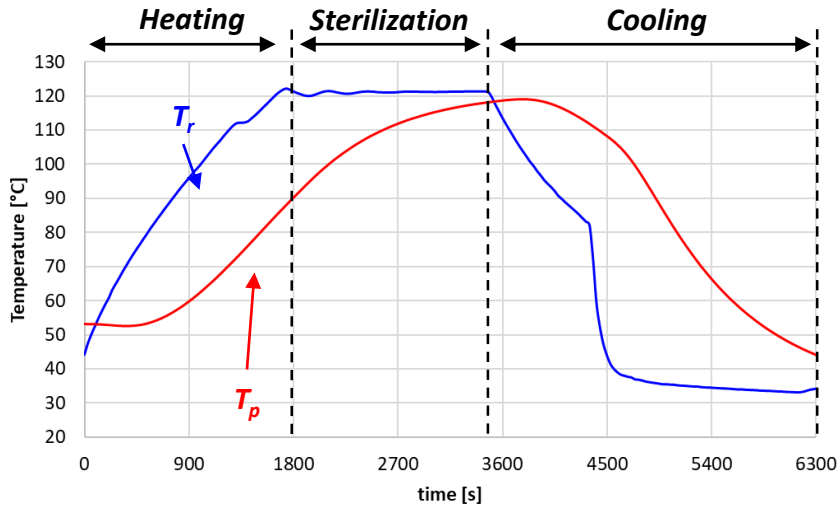
Blue line = Envelope of D-values as a function of the Temperature (T_i).

Z_{value} = temperature increase necessary to reduce the D_{value} by a logarithmic order



Heat penetration curve

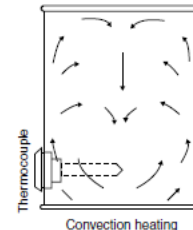
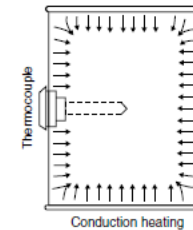
Typical heat treatment processes are made with **retort** and are usually composed by **3 steps** :



T_r = retort temperature.

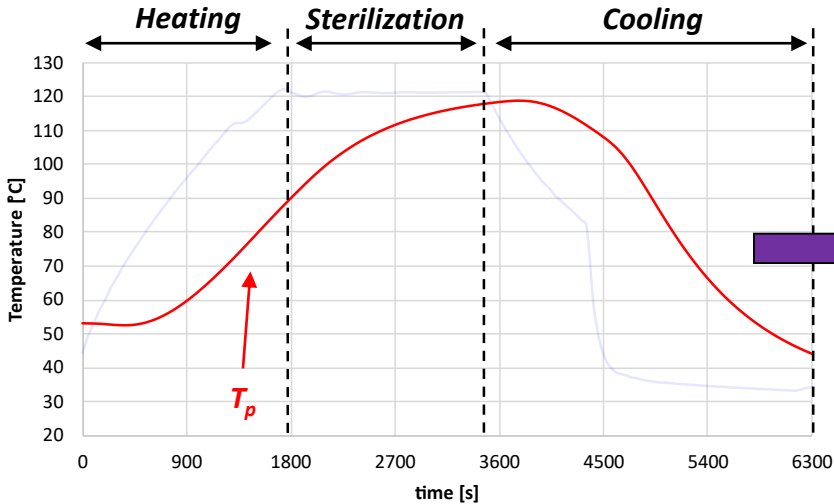
T_p = Coldest product point temperature

- The shape of the T_p curve can be significantly different as the location of the coldest point in the can, according to their physical state.



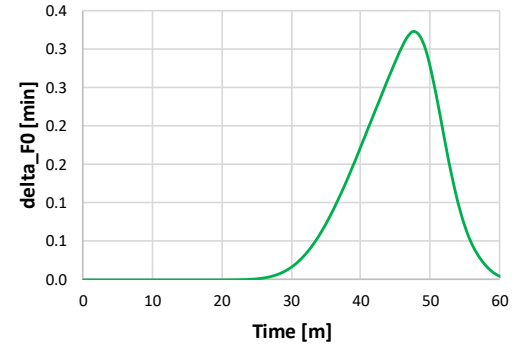
Lethality factor

By integrating the heating curve of the product, we can obtain the **lethality factor** F_{value} , define as the time to neutralize all bacteria for a given temperature.

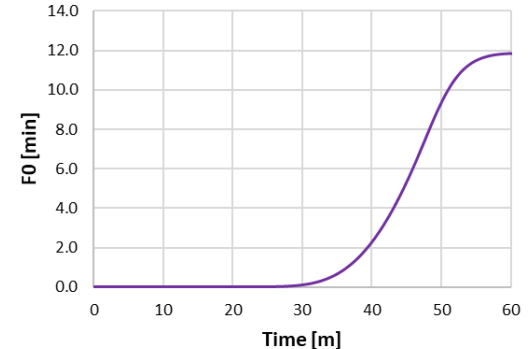


$$F_{value} = \int_0^t 10^{\frac{T_p(t) - T_{ref}}{Z_{ref}}} dt$$

T_{ref} = Reference temperature
 Z_{ref} = Z_{value} of the bacterium at T_{ref}
 $T_p(t)$ = product temperature behavior
 dt = heat treatment time



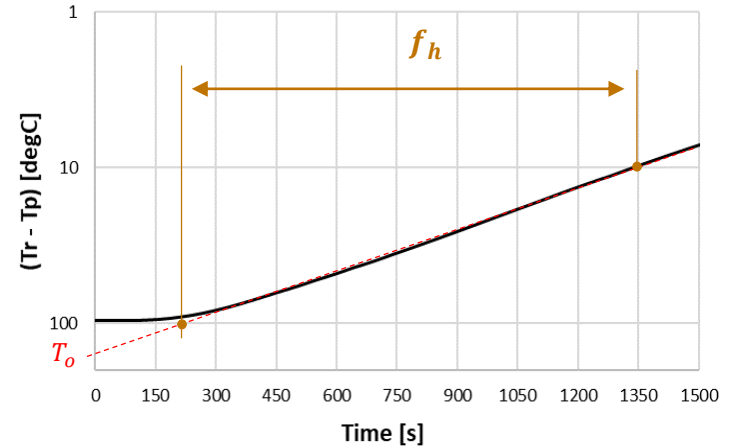
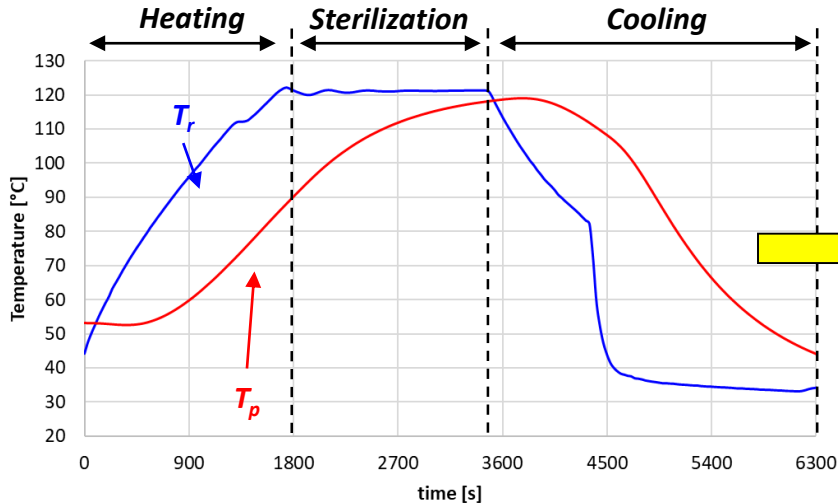
$$F_0 = F_{value} @ T_{ref} = 121^\circ C$$



- F_{value} = equivalent time to neutralize the same number of bacteria of an ideal thermal cycle at temperature T_{ref}

Other process parameters

Furthermore, considering the function $(T_r - T_p)$ in logarithmic scale, we obtain the plot in the right part.



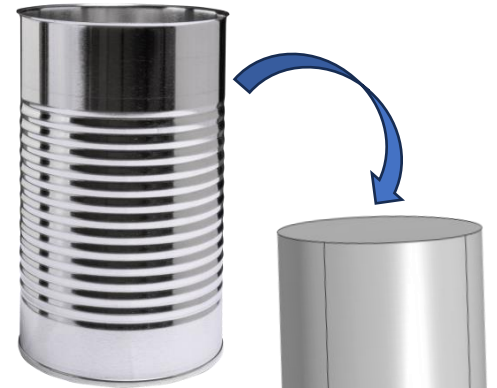
- The **delay factor** J_c is the initial lag time before the internal temperature of a product begins to rise significantly during heating.

$$J_c = \frac{T_r - T_o}{T_r - T_{init}}$$

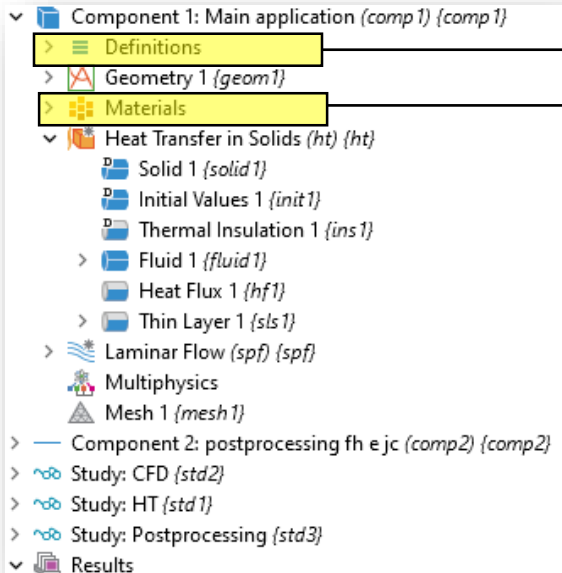
- The **penetration factor** f_h is the time necessary to reduce the difference between T_r and T_p by one logarithmic cycle.

Model implementation

- A thermo-fluid dynamic model in COMSOL environment was built to analyze the heat penetration inside the box.



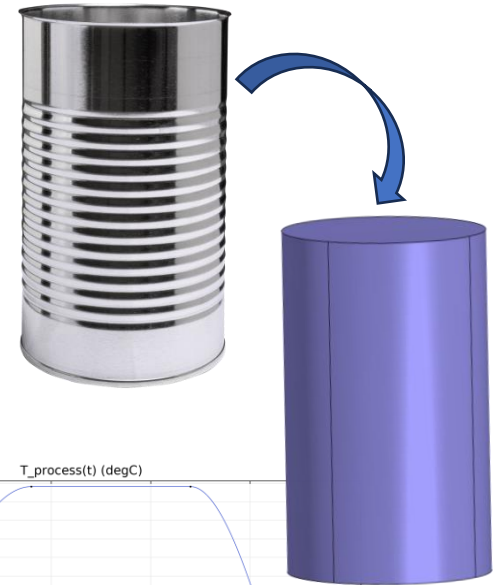
- Thermal properties of product are made of analytical equation refers to the nutritional values.



Name	Expression	Unit	De
rho_prot	$1.3299 \cdot 10^3 \cdot (-3) - 5.1840 \cdot 10^6 \cdot (-1) \cdot T \cdot d^{\wedge}1 / 11 [K]$		
k_prot	$1.7881 \cdot 10^4 \cdot (-1) + 1.1958 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 2.7178 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
Cp_prot	$2.0082 + 1.2089 \cdot 10^4 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 6.7036 \cdot 10^6 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
rho_fat	$9.2559 \cdot 10^3 \cdot (-2) - 4.1757 \cdot 10^6 \cdot (-1) \cdot T \cdot d^{\wedge}1 / 11 [K]$		
k_fat	$1.8071 \cdot 10^4 \cdot (-1) - 2.7694 \cdot 10^6 \cdot (-4) \cdot T \cdot d^{\wedge}1 - 1.7749 \cdot 10^9 \cdot (-7) \cdot T \cdot d^{\wedge}2$		
Cp_fat	$1.9842 + 1.4733 \cdot 10^4 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 4.8008 \cdot 10^6 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
rho_carbo	$1.5991 \cdot 10^3 \cdot (-3) - 3.1046 \cdot 10^6 \cdot (-1) \cdot T \cdot d^{\wedge}1 / 11 [K]$		
k_carbo	$2.0141 \cdot 10^4 \cdot (-1) - 1.3874 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 4.3312 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
Cp_carbo	$1.5468 + 1.9625 \cdot 10^4 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 5.9399 \cdot 10^6 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
rho_fiber	$1.3115 \cdot 10^3 \cdot (-3) - 3.6559 \cdot 10^6 \cdot (-1) \cdot T \cdot d^{\wedge}1 / 11 [K]$		
k_fiber	$1.8331 \cdot 10^4 \cdot (-1) + 1.2497 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 3.1683 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
Cp_fiber	$1.8459 + 1.8308 \cdot 10^4 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 4.6509 \cdot 10^6 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
rho_water	$9.9718 \cdot 10^3 \cdot (-2) + 3.1439 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 3.7574 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
k_water	$5.7109 \cdot 10^4 \cdot (-1) + 1.7625 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 6.7036 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
Cp_water	$4.1289 - 9.0864 \cdot 10^4 \cdot (-5) \cdot T \cdot d^{\wedge}1 + 5.4731 \cdot 10^7 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
rho_ash	$2.4238 \cdot 10^3 \cdot (-3) - 2.8063 \cdot 10^6 \cdot (-1) \cdot T \cdot d^{\wedge}1 / 11 [K]$		
k_ash	$3.2962 \cdot 10^4 \cdot (-1) + 1.4011 \cdot 10^6 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 2.9069 \cdot 10^9 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
Cp_ash	$1.0926 + 1.8898 \cdot 10^4 \cdot (-3) \cdot T \cdot d^{\wedge}1 - 3.6817 \cdot 10^6 \cdot (-6) \cdot T \cdot d^{\wedge}2$		
T d	T-273.15	K	

Model implementation

- **Conductive heat exchange** was solved inside the can (but also natural or forced heat exchange can be simulated, according to the physical states of the product and the specific treatment).



- **Convective heat flux** was imposed along the wall of the can, according to process temperature

Component 1: Main application (comp1) {comp1}

- Definitions
- Geometry 1 {geom1}
- Materials
- Heat Transfer in Solids (ht) {ht}
 - Solid 1 {solid1}
 - Initial Values 1 {init1}
 - Thermal Insulation 1 {ins1}
 - Fluid 1 {fluid1}
 - Heat Flux 1 {hf1}
 - Thin Layer 1 {sls1}
 - Laminar Flow (spf) {spf}
 - Multiphysics
 - Mesh 1 {mesh1}
- Component 2: postprocessing fh e jc (comp2) {comp2}
- Study: CFD {std2}
- Study: HT {std1}
- Study: Postprocessing {std3}
- Results

Heat Flux

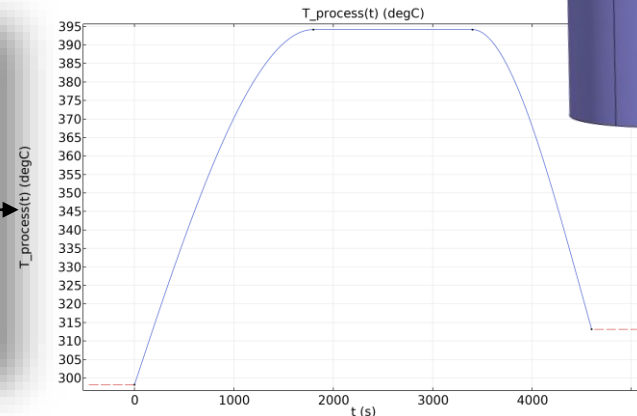
Flux type:
Convective heat flux

Heat transfer coefficient:
User defined

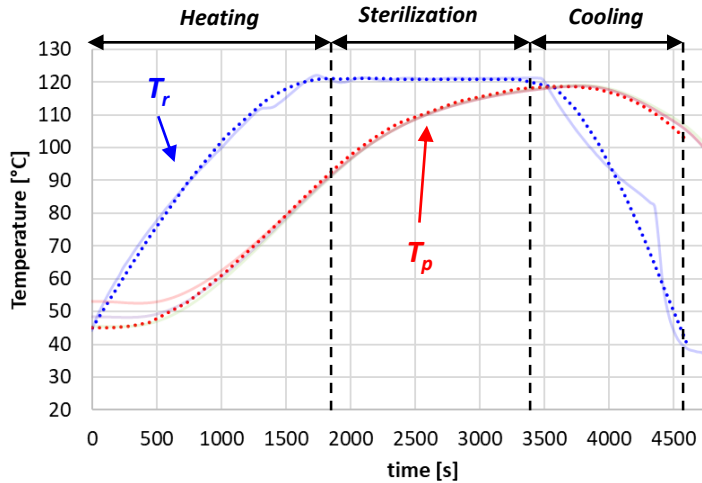
Heat transfer coefficient:
h h_conv W/(m²·K)

External temperature:
T_ext User defined

T_process(t) K

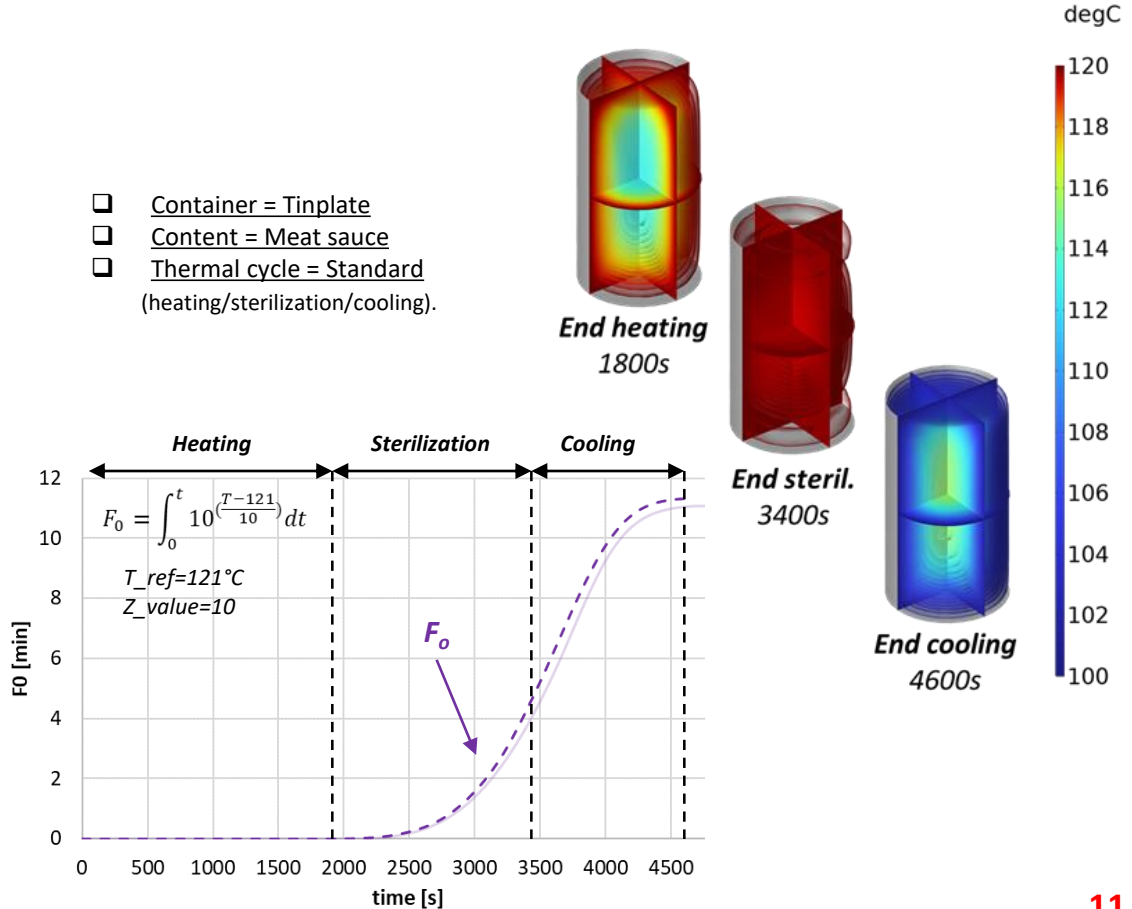


NUM-EXP Validation

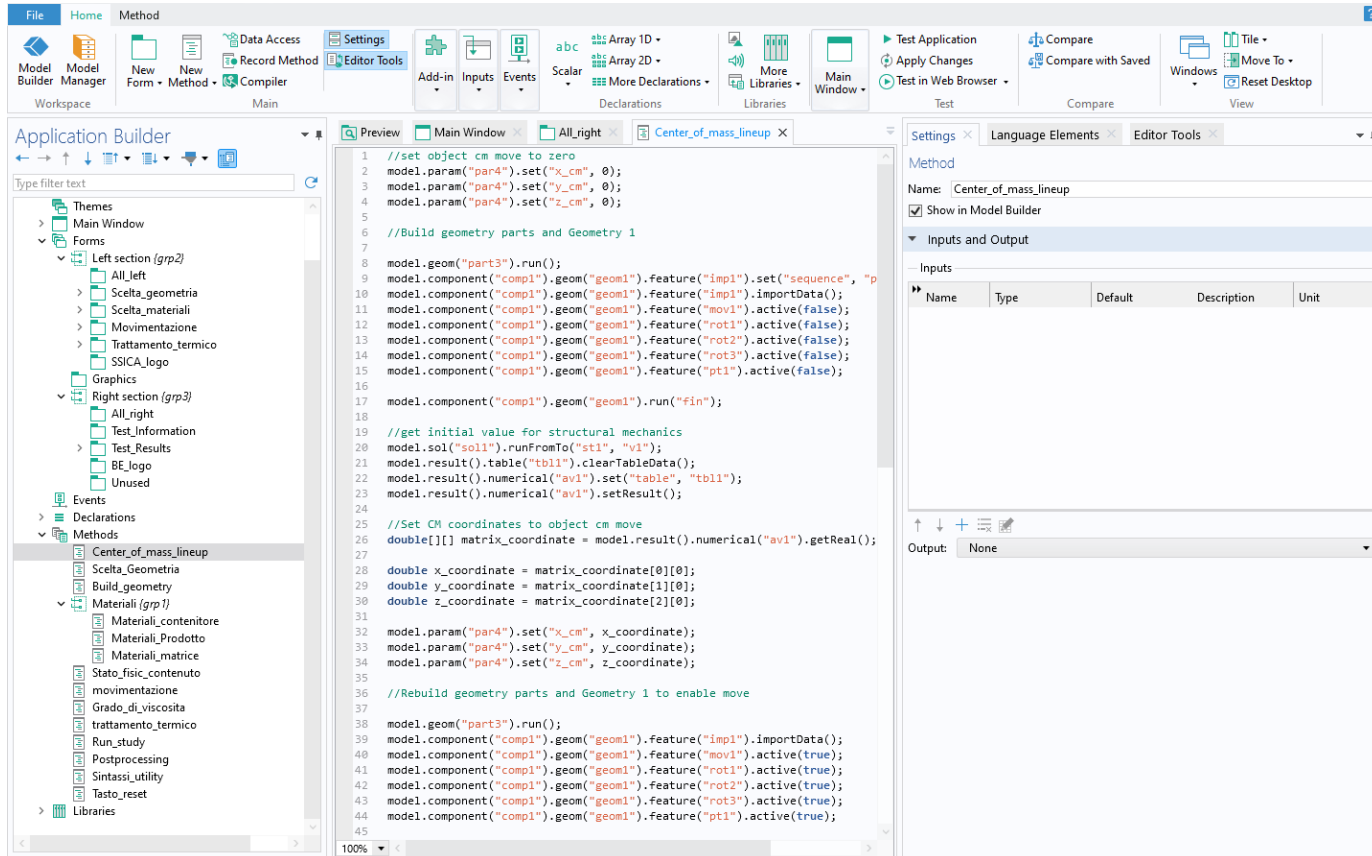


- Container = Tinplate
- Content = Meat sauce
- Thermal cycle = Standard (heating/sterilization/cooling).

- Checking out our numerical temperature profile with some experimental curves, we achieve a **very good agreement** also in the lethality factor behavior.



Forms and methods: to make the model user-friendly



The screenshot displays the ANSYS Workbench Application Builder interface. The main window shows a code editor with the following content:

```

1 //set object cm move to zero
2 model.param("par4").set("x_cm", 0);
3 model.param("par4").set("y_cm", 0);
4 model.param("par4").set("z_cm", 0);
5
6 //Build geometry parts and Geometry 1
7
8 model.geom("part3").run();
9 model.component("comp1").geom("geom1").feature("imp1").set("sequence", "p
10 model.component("comp1").geom("geom1").feature("imp1").importData();
11 model.component("comp1").geom("geom1").feature("mov1").active(false);
12 model.component("comp1").geom("geom1").feature("rot1").active(false);
13 model.component("comp1").geom("geom1").feature("rot2").active(false);
14 model.component("comp1").geom("geom1").feature("rot3").active(false);
15 model.component("comp1").geom("geom1").feature("pt1").active(false);
16
17 model.component("comp1").geom("geom1").run("fin");
18
19 //get initial value for structural mechanics
20 model.sol("sol1").runFromTo("st1", "v1");
21 model.result().table("tbl1").clearTableData();
22 model.result().numerical("av1").set("table", "tbl1");
23 model.result().numerical("av1").setResult();
24
25 //Set CM coordinates to object cm move
26 double[][] matrix_coordinate = model.result().numerical("av1").getResult();
27
28 double x_coordinate = matrix_coordinate[0][0];
29 double y_coordinate = matrix_coordinate[1][0];
30 double z_coordinate = matrix_coordinate[2][0];
31
32 model.param("par4").set("x_cm", x_coordinate);
33 model.param("par4").set("y_cm", y_coordinate);
34 model.param("par4").set("z_cm", z_coordinate);
35
36 //Rebuild geometry parts and Geometry 1 to enable move
37
38 model.geom("part3").run();
39 model.component("comp1").geom("geom1").feature("imp1").importData();
40 model.component("comp1").geom("geom1").feature("mov1").active(true);
41 model.component("comp1").geom("geom1").feature("rot1").active(true);
42 model.component("comp1").geom("geom1").feature("rot2").active(true);
43 model.component("comp1").geom("geom1").feature("rot3").active(true);
44 model.component("comp1").geom("geom1").feature("pt1").active(true);
45

```

The right-hand pane shows the 'Method' configuration for 'Center_of_mass_lineup'. It includes a 'Show in Model Builder' checkbox and an 'Inputs and Output' table:

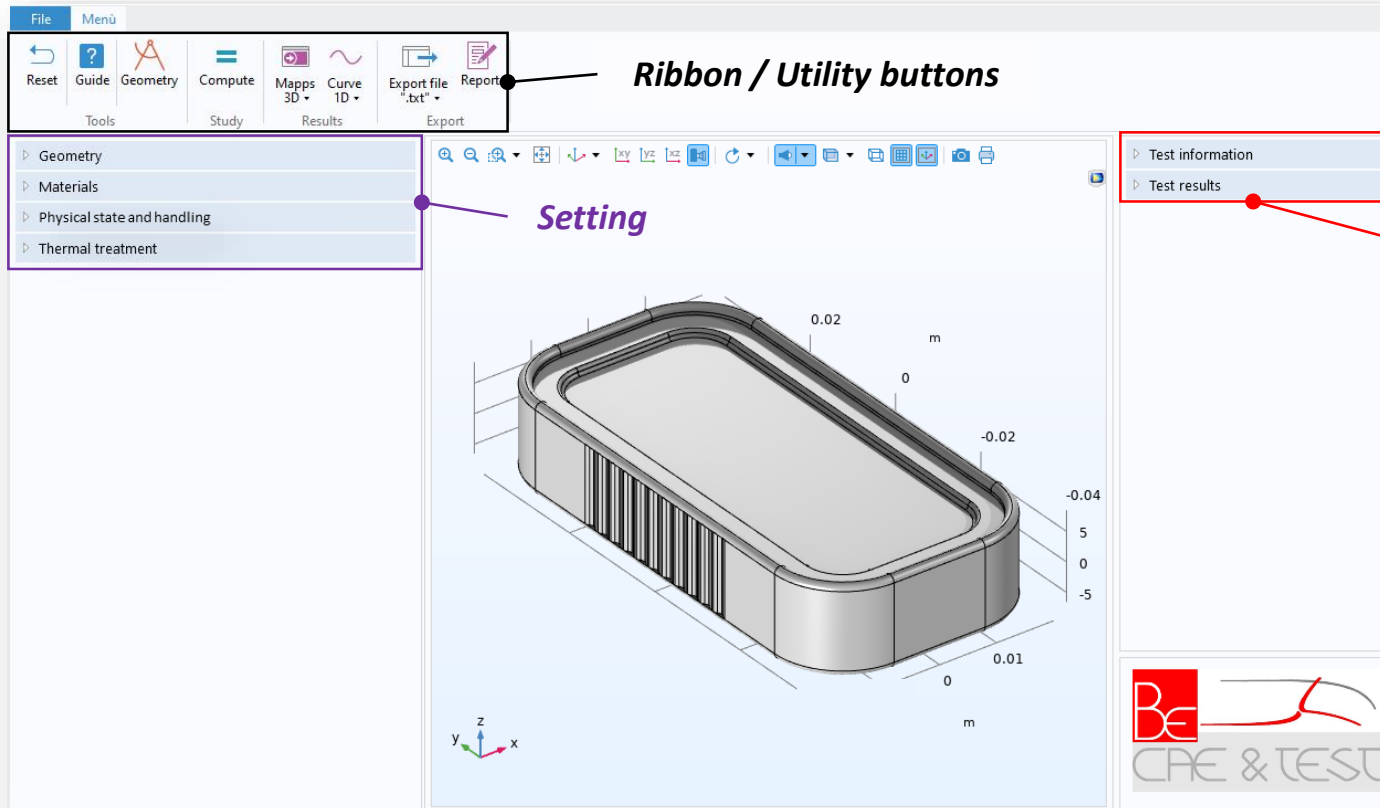
Name	Type	Default	Description	Unit
None				

- **Application builder** interface was used to design a **simplified graphics** interface for all customers, from data entry to the result.

- Several **methods** and **form** have been implemented to **manage** the various **features** of the application.

Let's take a closer look at how the app works in the next slides.

App UI



Ribbon / Utility buttons

Setting


*Post-processing
and test
information*

App UI: Geometry

Choose the **shape** of the **container** and the main **dimensions**.


> Geometry
 > Materials
 > Physical state and handling
 > Thermal treatment

▼ Geometry
Shape:


 Build geometry

Position: *x_dir* *y_dir* *z_dir*

Translation	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	[mm]
Rotation	<input type="text" value="0"/>	<input type="text" value="-90"/>	<input type="text" value="0"/>	[deg]

 Reposition

Shape: Cylinder
 Diameter: 45 [mm]
 Height: 30 [mm]
 Thickness: 1 [mm]

Shape: Parallelepiped
 Length: 10 [mm]
 Width: 5 [mm]
 Height: 30 [mm]
 Thickness: 1 [mm]


Shape: Imped geometry
 Browse...



App UI: Geometry


Choose the **shape** of the **container** and the main **dimensions**.


> Geometry
 > Materials
 > Physical state and handling
 > Thermal treatment


Geometry
Shape:
1) Select a shape for the geometry

Position:

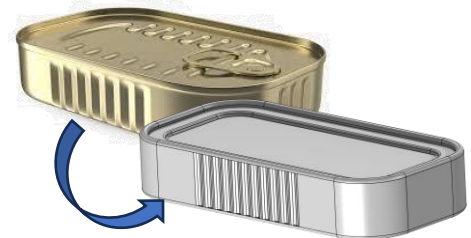
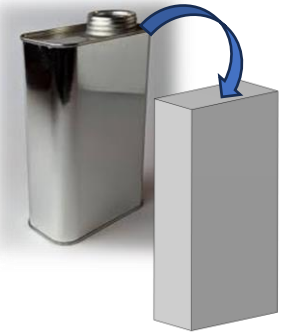
	<i>x_dir</i>	<i>y_dir</i>	<i>z_dir</i>	
Translation	0	0	0	[mm]
Rotation	0	-90	0	[deg]

Shape: Import geometry



Shape: Cylinder
 Diameter:
 Height:
 Thickness:


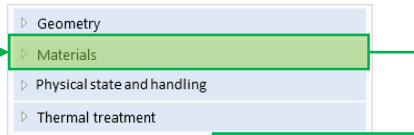
Shape: Parallelepiped
 Length:
 Width:
 Height:
 Thickness:




2) Positioning and orientation

- **Rotate** the container if is necessary

App UI: Materials



▼ Materials

Container:

Material:

Content:

Product:

Matrix:

Insert **thermal properties** through a pre-implemented materials library or a “user defined” settings.

Content:

Product:

Mushrooms
Onions
Pepper
Tomato
Corn
Rice
Beans
Chickpeas
Lentils
Salmon
Tuna
Boiled risot
Peach
Pineapple
Orange juice

User defined - constant parameters

User defined - f(T)

User defined - nutritional values

Pre-implemented materials library

Product: User defined - constant parameters

rho_product: [kg/m^3]

k_product: [W/m^2K]

Cp_product: [J/kg^oK]

Product: User defined - f(T)

rho_product(T):

k_product(T):

Cp_product(T):

Product: User defined - Nutritional values

Nutritional values per 100g of drained product:

Proteins:	23.4
Carbohydrates:	0
Fats:	4.9
Fibers:	0
Acid:	1.2

- “Constant parameters” and “f(T)” settings are like the previous slide.

- “Nutritional values” setting estimate thermal properties of products starting from their nutritional values.

Matrix:

Matrix: Oil matrix

Product/matrix ratio:

Total net weight (g) = 80 Drained weight (g) = 56

Matrix:

Matrix: Water matrix

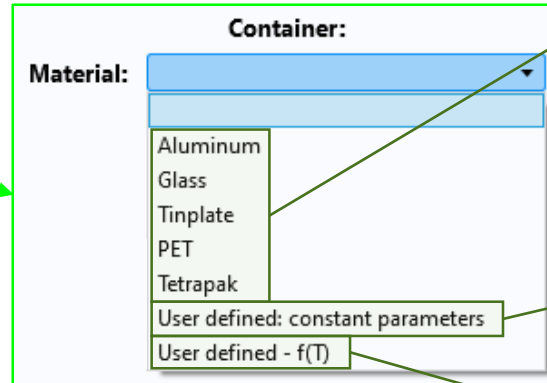
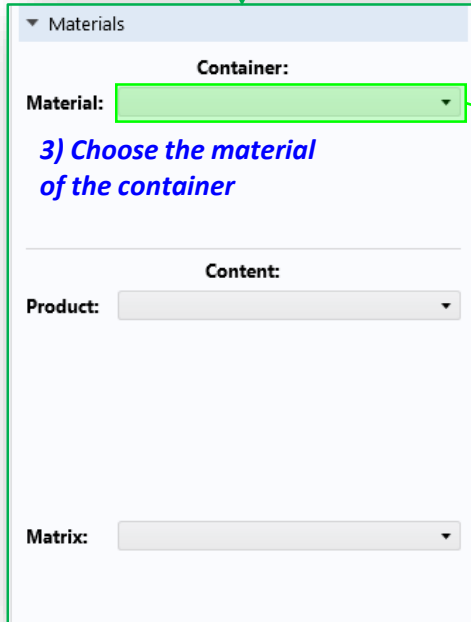
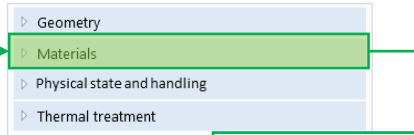
Product/matrix ratio:

Total net weight (g) = 80 Drained weight (g) = 56

- If product is packed in a fluid matrix (such as oil or Brine), it is necessary to specify the total weight and the net weight of the can; the app will calculate averages thermal properties between matrix and product as a function of the specified weights.

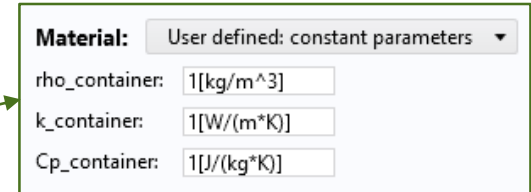
App UI: Materials / Container

Allocate **thermal properties** through a pre-implemented materials library or a “user defined” settings.

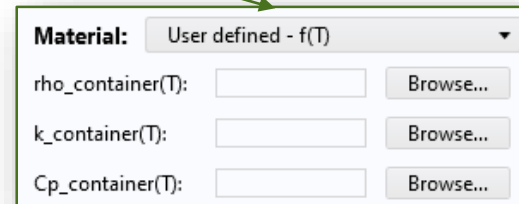


Pre-implemented materials library

- In the “constant parameters” setting, user must **import** thermal properties value as a **single value**.



- In the “f(T)” parameters setting, user must **import** thermal properties as **curves functions of temperature**.



App UI: Materials / Content

Insert **thermal properties** through a pre-implemented materials library or a “user defined” settings.

▸ Geometry
 ▸ **Materials**
 ▸ Physical state and handling
 ▸ Thermal treatment

▾ Materials
Container:
 Material:

Content:
 Product:

4) Choose the material of the content

Matrix:

Content:

Product:

- Mushrooms
- Onions
- Pepper
- Tomato
- Corn
- Peas
- Beans
- Chickpeas
- Lentils
- Salmon
- Tuna
- Boiled meat
- Peach
- Pineapple
- Orange juice
- User defined - constant parameters
- User defined - f(T)
- User defined - Nutritional values

Pre-implemented materials library

Product: User defined - constant parameters

rho_product: [kg/m^3]
 k_product: [W/(m*K)]
 Cp_product: [J/(kg*K)]

Product: User defined - f(T)

rho_product(T): Browse...
 k_product(T): Browse...
 Cp_product(T):

Product: User defined - Nutritional values

Nutritional values per 100g of drained product:

Proteins:	<input type="text"/> 23.3
Carbohydrates:	<input type="text"/> 0
Fats:	<input type="text"/> 4.9
Fibers:	<input type="text"/> 0
Ash:	<input type="text"/> 1.2

- “Constant parameters” and “f(T)” settings are like the previous slide.
- “Nutritional values” setting **estimate** thermal properties of products starting from their **nutritional values**.

Matrix: Oil matrix

Product/matrix ratio:

Matrix: Water matrix

Total net weight (g): 80 Drained weight (g): 56

Product/matrix ratio:

Total net weight (g): 80 Drained weight (g): 56

If product is packed in a fluid matrix (such as oil or Brine), it is necessary to specify the total weight and the net weight of the can; the app will calculate averages thermal properties between matrix and product as a function of the specified weights.

App UI: Materials / Content

Insert **thermal properties** through a pre-implemented materials library or a “user defined” settings.

- ▷ Geometry
- ▷ **Materials**
- ▷ Physical state and handling
- ▷ Thermal treatment

▼ Materials

Container:

Material:

Content:

Product:

Matrix:

5) Specify if the content has biphasic states

Content:

Product:

- Mushrooms
- Onions
- Pepper
- Tomato
- Corn
- Peas
- Beans
- Chickpeas
- Lentils
- Salmon
- Tuna
- Boiled meat
- Peach
- Pineapple
- Orange juice

User defined - constant parameters

User defined - f(T)

User defined - Nutritional values

Pre-implemented materials library

Product: User defined - constant parameters

rho_product: [kg/m^3]

k_product: [W/(m*K)]

Cp_product: [J/(kg*K)]

Product: User defined - f(T)

rho_product(T): Browse...

k_product(T): Browse...

Cp_product(T):

Product: User defined - Nutritional values

Nutritional values per 100g of drained product:

Proteins:	<input type="text"/> 23.3
Carbohydrates:	<input type="text"/> 0
Fats:	<input type="text"/> 4.9
Fibers:	<input type="text"/> 0
Ash:	<input type="text"/> 1.2

- “Constant parameters” and “f(T)” settings are like the previous slide.
- “Nutritional values” setting *estimate* thermal properties of products starting from their **nutritional values**.

Matrix:

Matrix: Oil matrix

Product/matrix ratio:

Total net weight [g]: 80 Drained weight[g]: 56

Matrix: Water matrix

Product/matrix ratio:

Total net weight [g]: 80 Drained weight[g]: 56

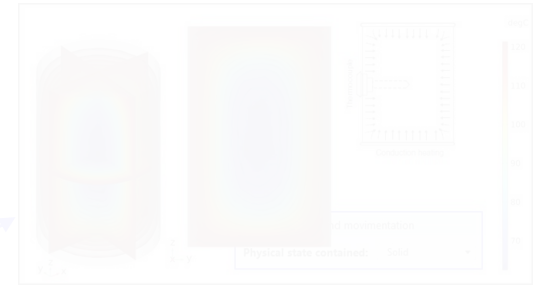
- **If product is packed in a fluid matrix (such as oil or Brine), it is necessary to specify the total weight and the net weight of the can; the app will calculate averages thermal properties between matrix and product as a function of the specified weights.**

App UI: Physical state

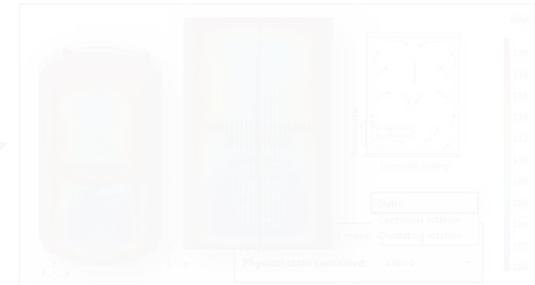
- ▷ Geometry
- ▷ Materials
- ▷ Physical state and handling
- ▷ Thermal treatment

Specify the **physical state** of the content. This will identify the **numerical model** for the **heat penetration**.

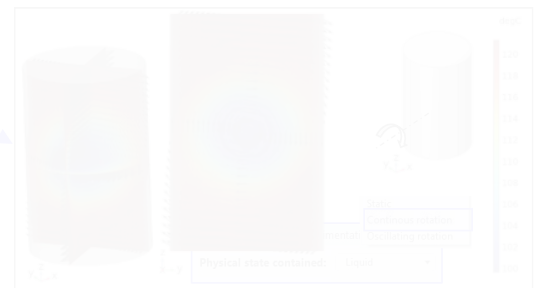
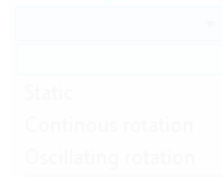
- “Solid” state simulate only a conductive heat exchange inside the can.



- “Liquid” state simulate both convective and conductive heat exchange inside the can.



- Specify the type of handling to change the convection model:
 - Static=natural convection flow;
 - Rotation=forced rotated flow;



- Since we are solving fluid dynamics, we must specify the degree of viscosity of the product



App UI: Physical state

- Geometry
- Materials
- Physical state and handling
- Thermal treatment

Specify the **physical state** of the content. This will identify the **numerical model** for the **heat penetration**.

- “Solid” state simulate conductive heat exchange inside the can.

▼ Physical state and handling

Physical state contained: ▼

▼ Physical state and handling

Physical state contained: Solid ▼

- “Liquid” state simulate both convective and conductive heat exchange inside the can.

▼ Physical state and handling

Physical state contained: Liquid

Grade of viscosity:

Type of movimentation:

- Specify the type of handling to change the convection model:
 - Static=natural convection flow;
 - Rotation=forced rotated flow;

- Since we are solving fluid dynamics, we must specify the degree of viscosity of the product

Water

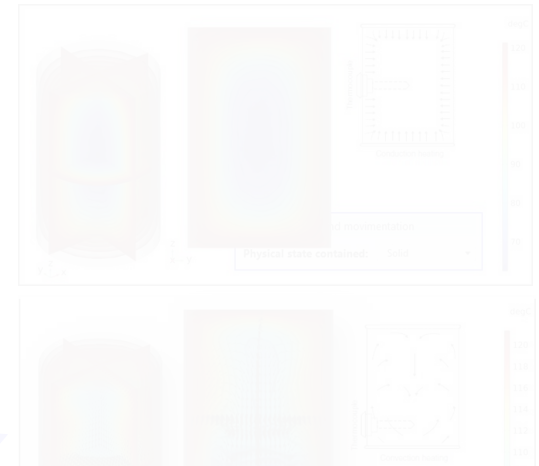
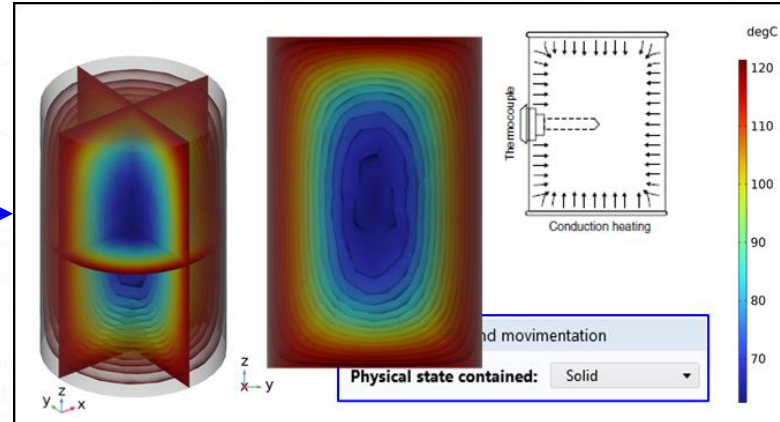
Fruit juice 40%

Fruit juice 50%

Fruit juice 60%

Oil

User defined(T)



App UI: Physical state

- ▷ Geometry
- ▷ Materials
- ▷ Physical state and handling
- ▷ Thermal treatment

Physical state and handling

Physical state contained: Liquid

6) Choose an appropriate numerical approach

- “Liquid” state simulate convective heat exchange inside the can.

Physical states and handling

Content physical state: Liquid

Grade of viscosity: Static

Handling: Static

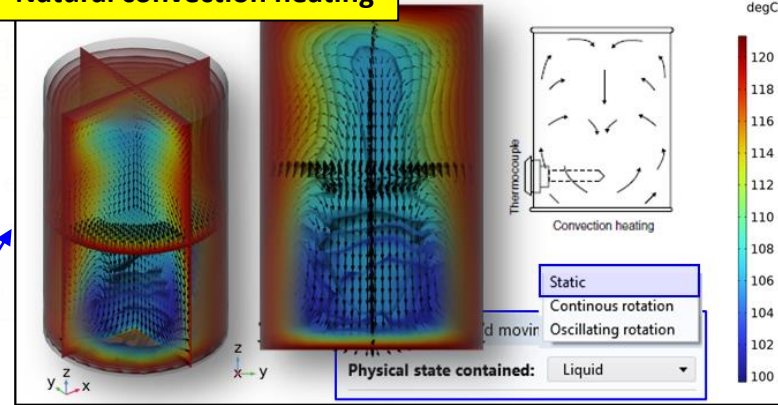
- Static
- Continous rotation
- Oscillating rotation

Specify the type of handling to change the convection model:

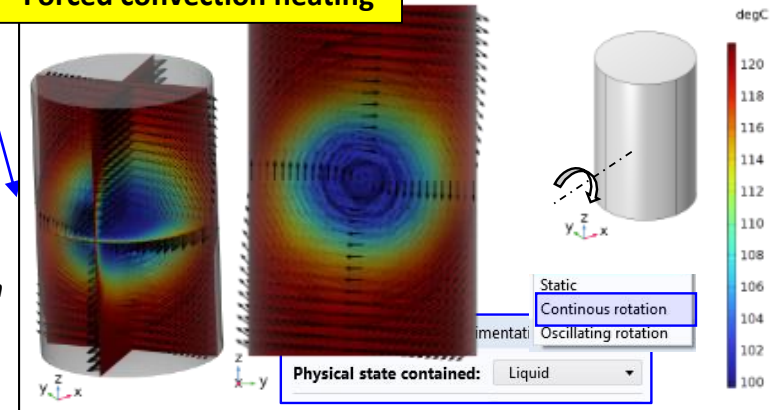
- Static=natural convection fluid flow;
- Rotation=forced rotated fluid flow;

- Water
- Fruit juice 40%
- Fruit juice 50%
- Fruit juice 60%
- Oli
- User defined(T)

Natural convection heating



Forced convection heating



App UI: Thermal treatment

- ▷ Geometry
- ▷ Materials
- ▷ Physical state and handling
- ▷ Thermal treatment

Specify the **thermal cycle**. Choose between "standard" or "user defined" mode.

Thermal treatment

Initial product temperature: 25[degC]

Thermal treatment: Standard

7) Choose a thermal treatment.

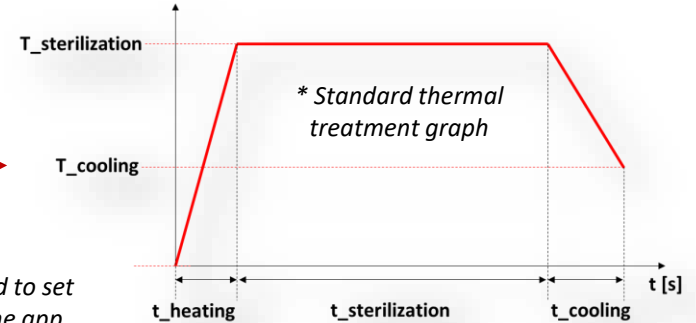
Lethality factor (F) parameters:

T_{value} [degC] 121

Z_{value} [-] 10

Thermal treatment: Standard

	Heating	Sterilization	Cooling
Temperature	25[degC]	121[degC]	40[degC]
Time	1800[s]	1600[s]	1200[s]

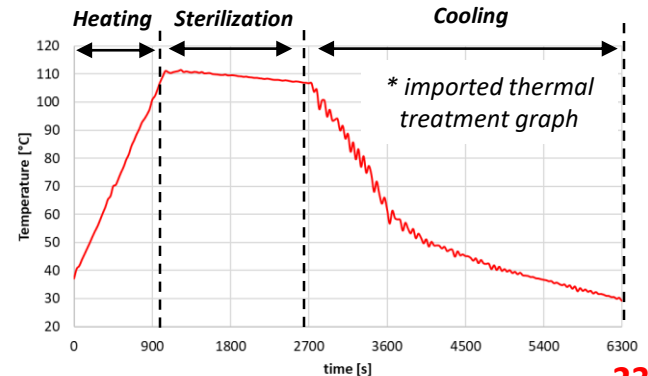


- In the "standard" treatment setting, we need to set the limit values of the thermal profile, and the app will create the curve as in the plot above.

Thermal treatment: User defined(T)

Browse...

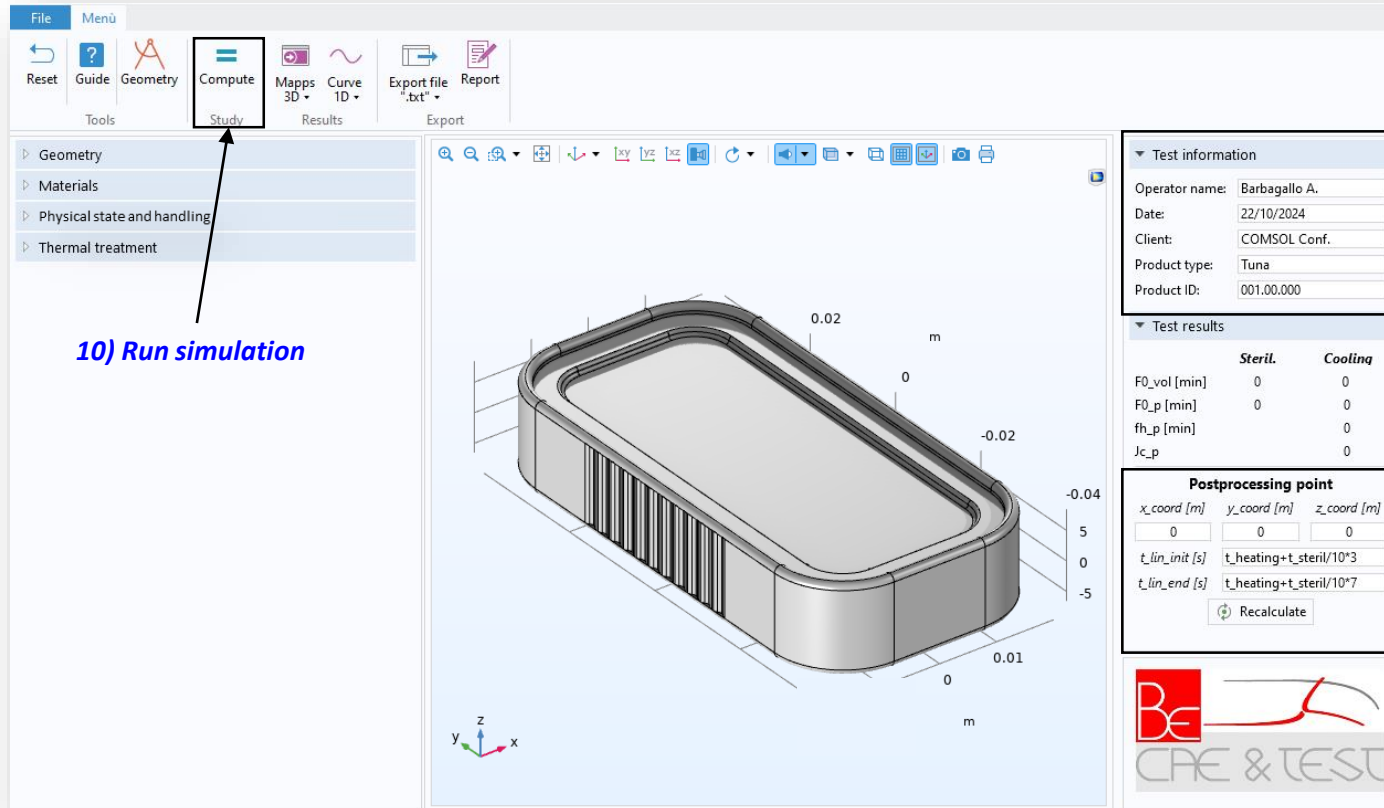
	Heating	Sterilization	Cooling
Time	1800[s]	1600[s]	1200[s]



8) Specify the characteristics of the bacterium we want to neutralize.

- Use the "user defined" setting to import a particular thermal curve. In this case, we need only specify the duration time of the three step of the treatment.

App UI: start simulation



The screenshot shows the COMSOL Multiphysics software interface. The top toolbar has the 'Compute' button highlighted with a red box and an arrow pointing to it. The left sidebar shows a tree view with 'Study' selected. The main workspace displays a 3D model of a rectangular container with dimensions 0.02 m by 0.01 m and a height of 0.04 m. The right sidebar contains the 'Test information' and 'Postprocessing point' sections.

Test information

Operator name:	Barbagallo A.
Date:	22/10/2024
Client:	COMSOL Conf.
Product type:	Tuna
Product ID:	001.00.000

Test results

	Steril.	Cooling
F0_vol [min]	0	0
F0_p [min]	0	0
fh_p [min]	0	0
Jc_p	0	0

Postprocessing point

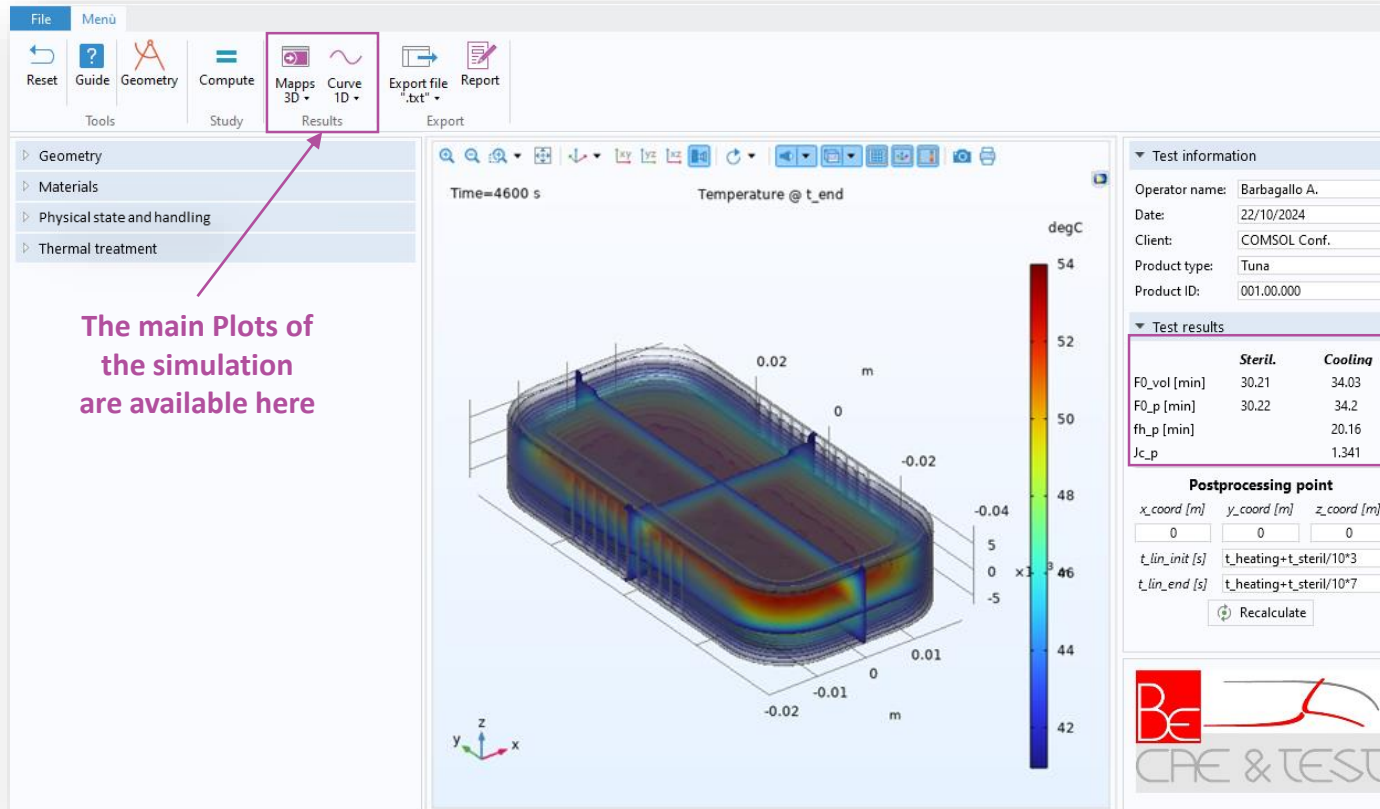
x_coord [m]	y_coord [m]	z_coord [m]
0	0	0
t_lin_init [s]	t_heating+t_steril/10*3	
t_lin_end [s]	t_heating+t_steril/10*7	

Recalculate

10) Run simulation

9) Fill the test information section and enter the coordinates of the point you want to analyse (by default the origin).

App UI: Results



File Menù

Reset Guide Geometry Compute Mapps 3D - Curve 1D - Export file ".txt" - Report

Tools Study Results Export

Geometry

Materials

Physical state and handling

Thermal treatment

Time=4600 s Temperature @ t_end

degC

54

52

50

48

46

44

42

0.02 m

0

-0.02

-0.04

5

0

-5

0.01 m

-0.02

-0.01

0

z

y

x

Test information

Operator name: Barbagallo A.

Date: 22/10/2024

Client: COMSOL Conf.

Product type: Tuna

Product ID: 001.00.000

Test results


	Steril.	Cooling
F0_vol [min]	30.21	34.03
F0_p [min]	30.22	34.2
fh_p [min]		20.16
Jc_p		1.341

Postprocessing point

x_coord [m]	y_coord [m]	z_coord [m]
0	0	0

t_lin_init [s]	t_heating+t_steril/10*3
t_lin_end [s]	t_heating+t_steril/10*7

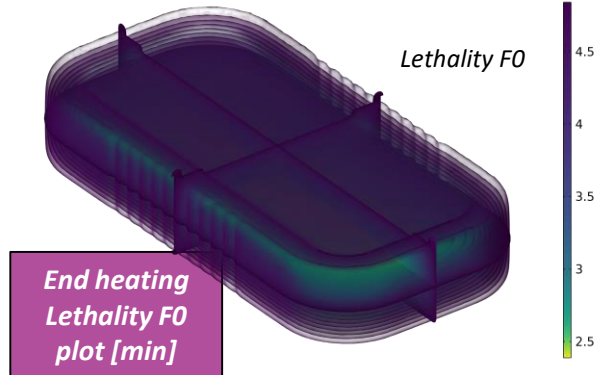
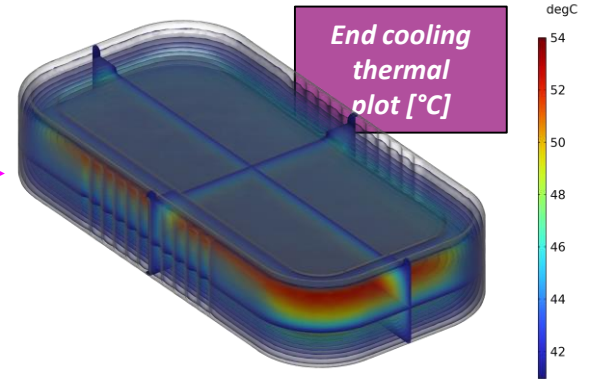
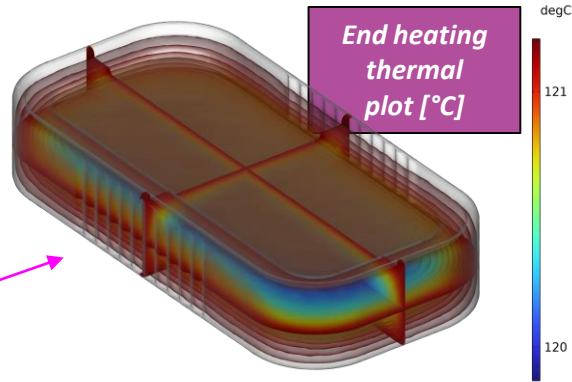
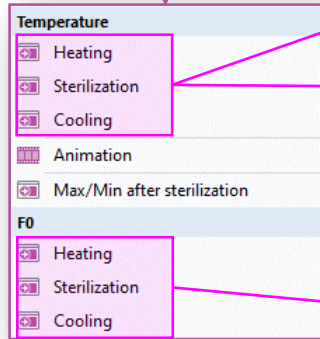
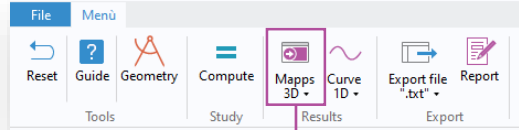
Recalculate



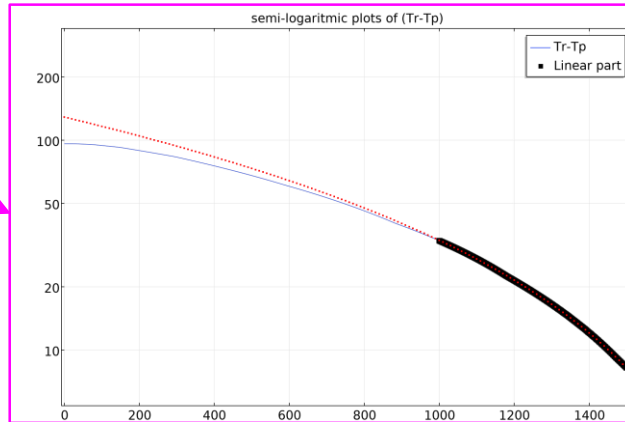
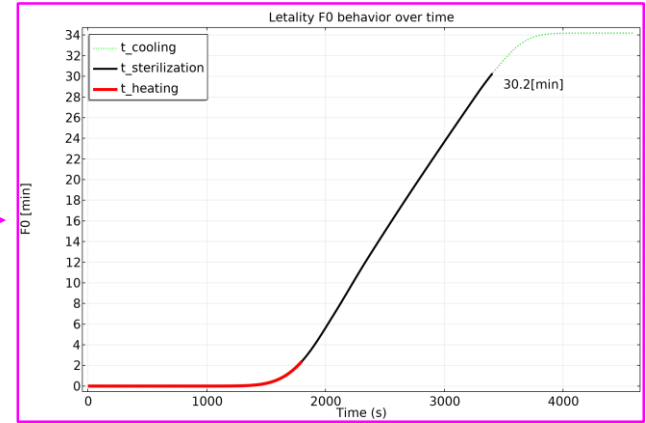
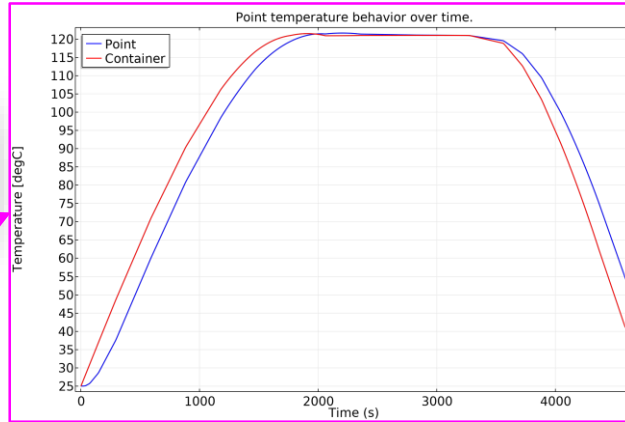
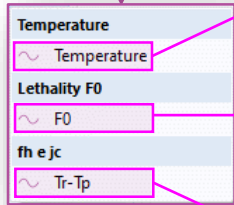
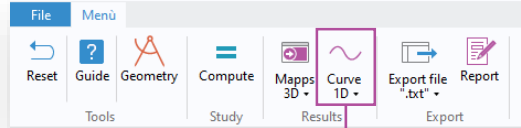
The main Plots of the simulation are available here

The main results of the simulation are shown here

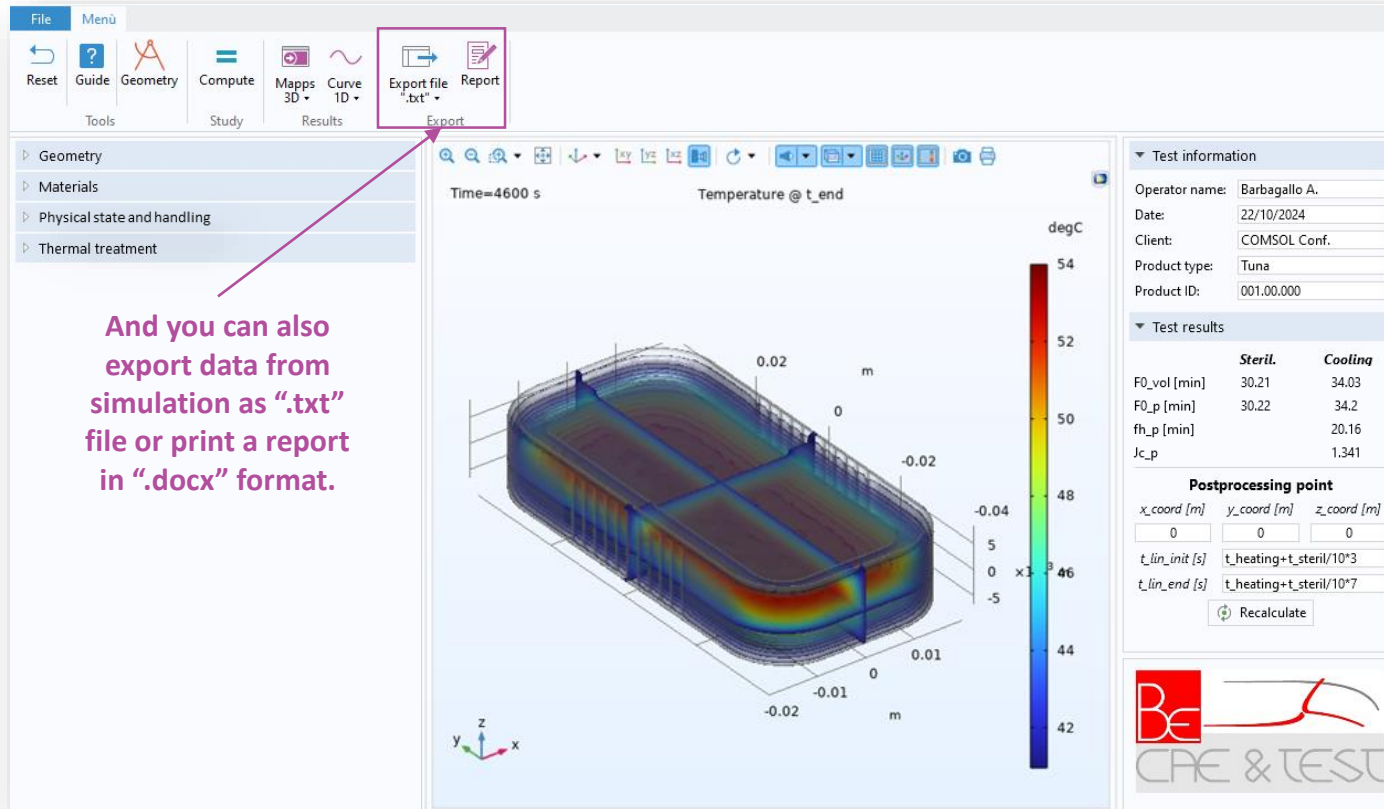
App UI: 3D Results



App UI: 1D results



App UI: Export



File Menù

Reset Guide Geometry Compute Maps 3D Curve ID Export file ".txt" Report

Tools Study Results Export

Geometry

Materials

Physical state and handling

Thermal treatment

Time=4600 s Temperature @ t_end

degC

54

52

50

48

46

44

42

0.02 m

0

-0.02

-0.04

5

0

-5

0.01 m

-0.02

-0.01

0

0.01

z

y

x

Test information

Operator name: Barbagallo A.

Date: 22/10/2024

Client: COMSOL Conf.

Product type: Tuna

Product ID: 001.00.000

Test results


	Steril.	Cooling
F0_vol [min]	30.21	34.03
F0_p [min]	30.22	34.2
fh_p [min]		20.16
Jc_p		1.341

Postprocessing point

x_coord [m]	y_coord [m]	z_coord [m]
0	0	0

t_lin_init [s]	t_heating+t_steril/10*3
t_lin_end [s]	t_heating+t_steril/10*7

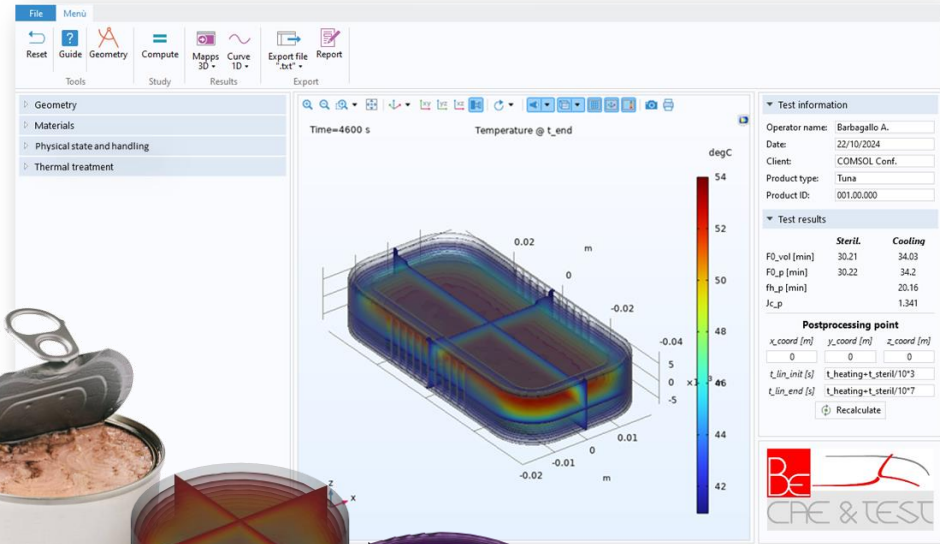
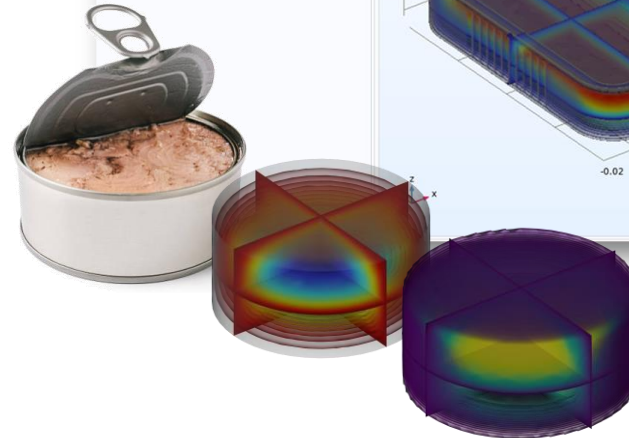
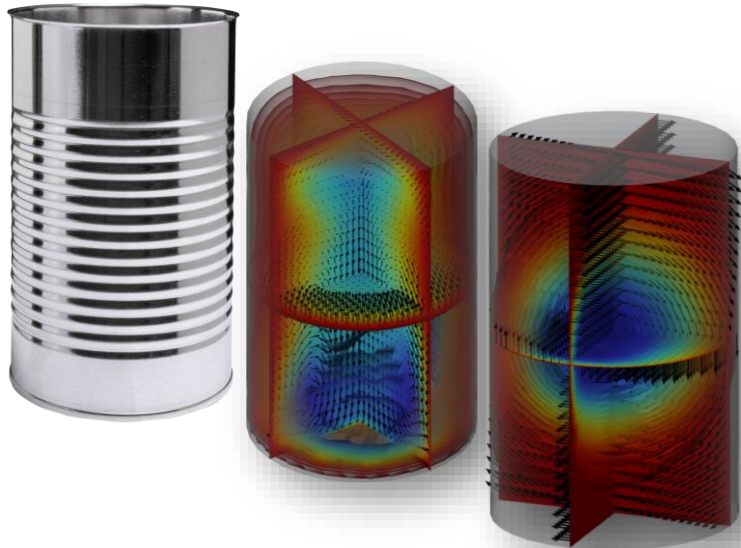
Recalculate



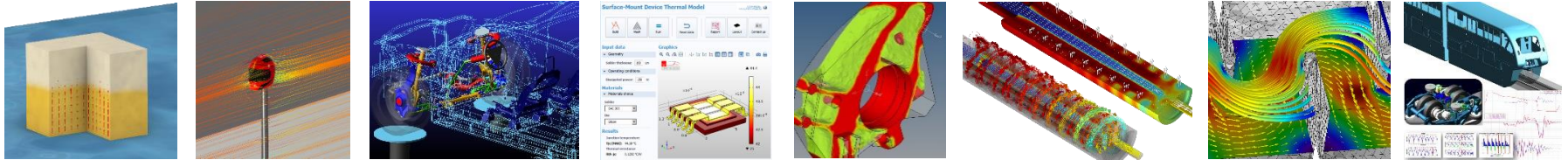
And you can also export data from simulation as ".txt" file or print a report in ".docx" format.




To summarize...

- Several protocols and industrial treatment are made up to **guarantee food safety** and preserve its quality.
- With this app, users can easily predict:
 1. the **bacteria reduction** following a heat treatment;
 2. the **efficiency** of the process;
 3. the **heat distribution** inside the can during the cycle.



Thank you all for your attention!



-  Viale Africa, 170 A - 95129 Catania (CT)
-  Via Toscana, 104 - 41053 Maranello (MO)
-  Calle Impresores, 20 - 28660 Boadilla del Monte (Madrid)

 www.be-caetest.it

 info@be-caetest.it

 +39 095 216 64 26

BE CAE & Test s.r.l.



BE CAE & Test S.r.l.
Be challenging, be smart: BE CAE & Test!

