

Součinnost nástroje MATLAB & Simulink s nástrojem fyzikálního modelování COMSOL Multiphysics

Model Controller

Setkání uživatelů COMSOL 2016

Pavlov (26. a 27.) května 2016

Jiří Marek



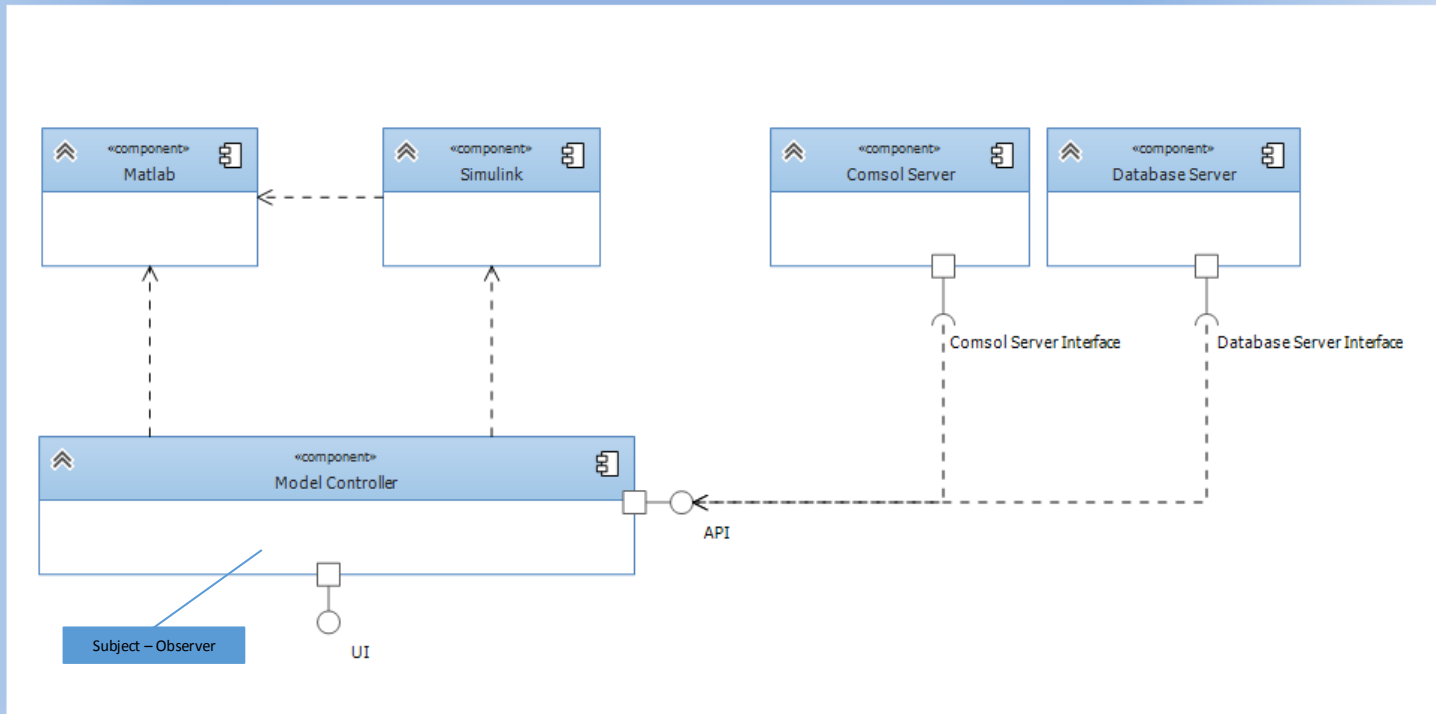
Vojenský výzkumný ústav, s. p.

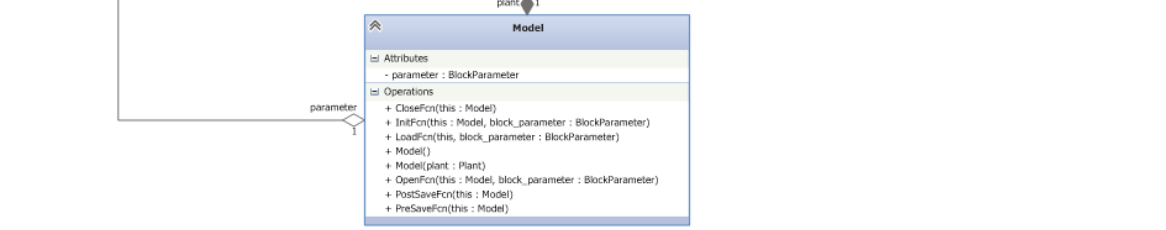
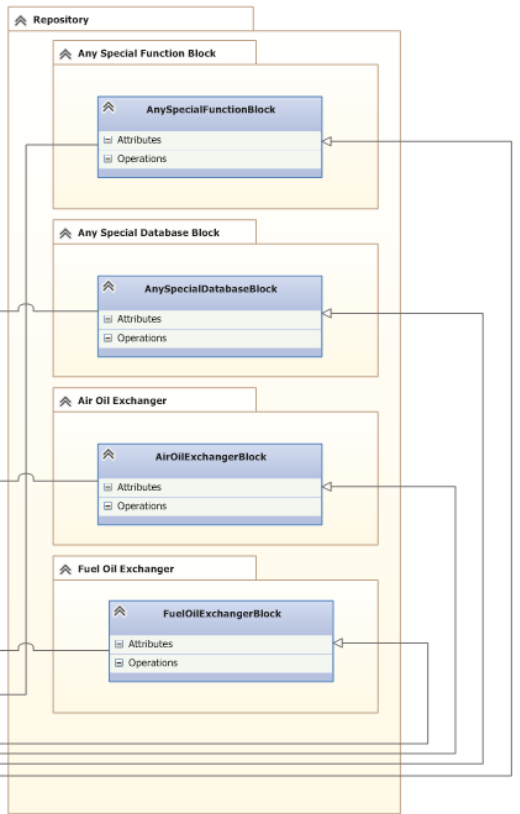
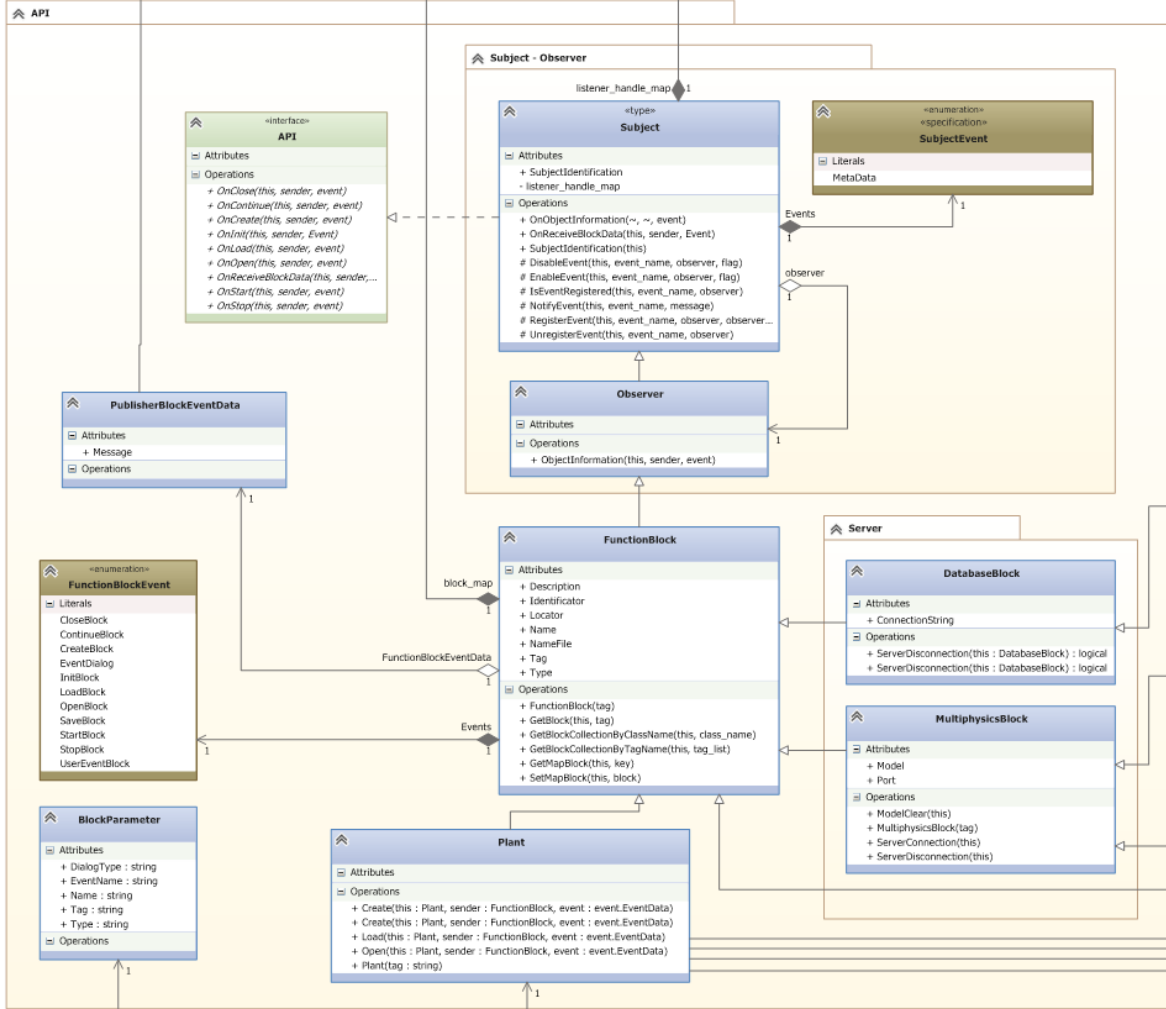
Veslařská 230, 637 00 B r n o

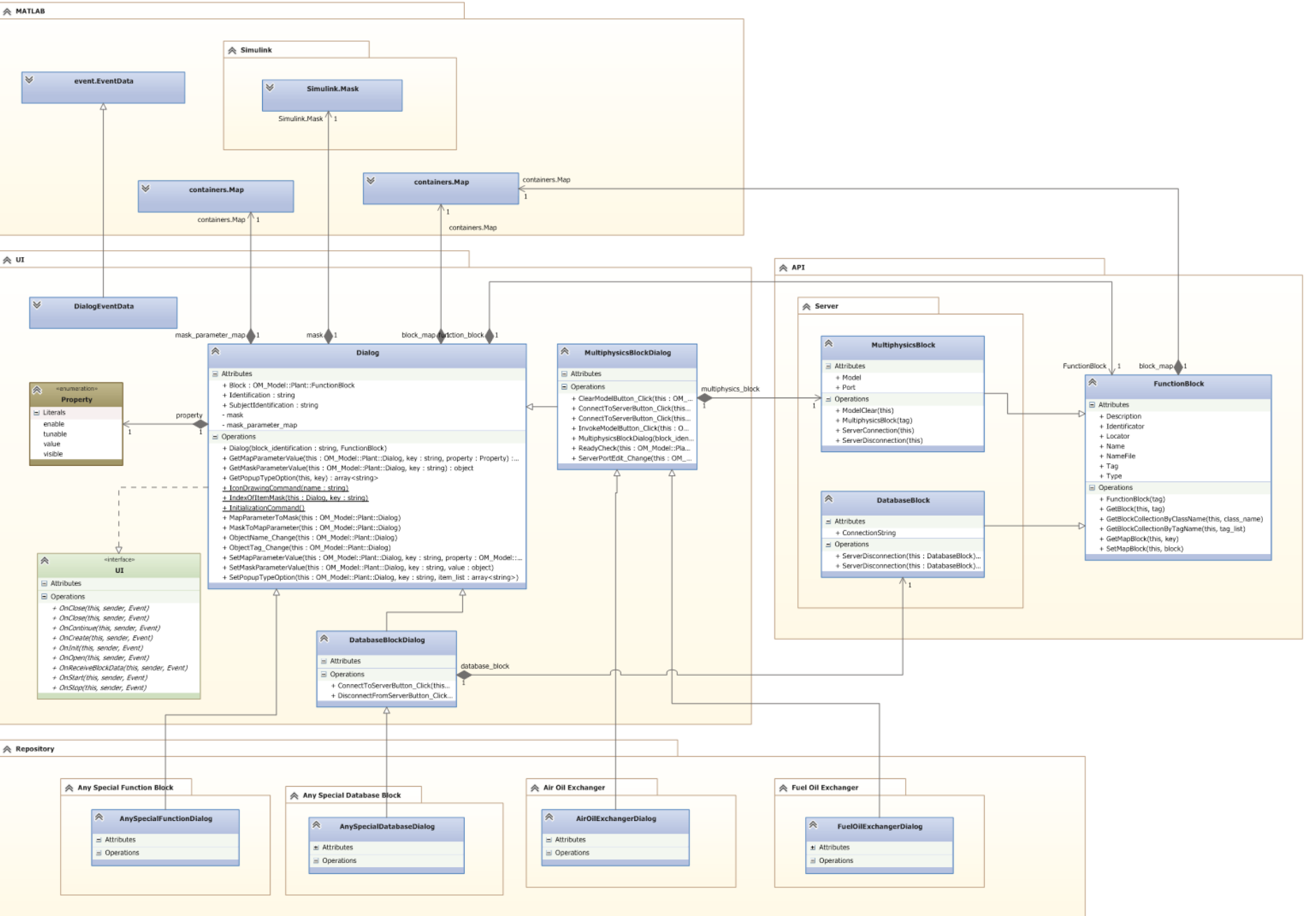
Tel: +420 543 562 101 Fax: +420 543 562 100

www.vvubrn.cz

Model Controller





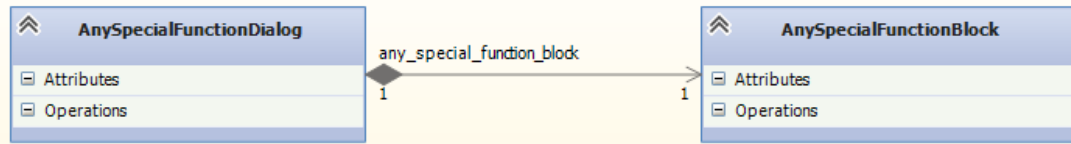


N

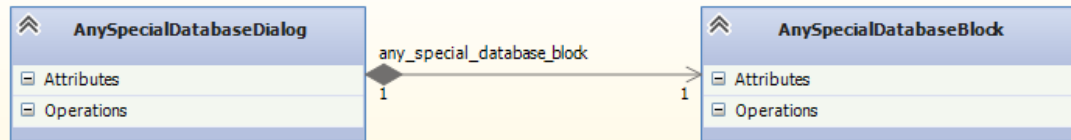
y

Repository

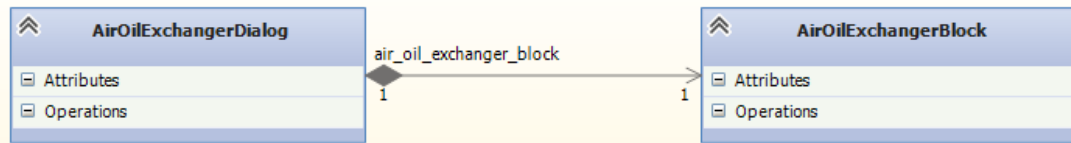
Any Special Function Block



Any Special Database Block

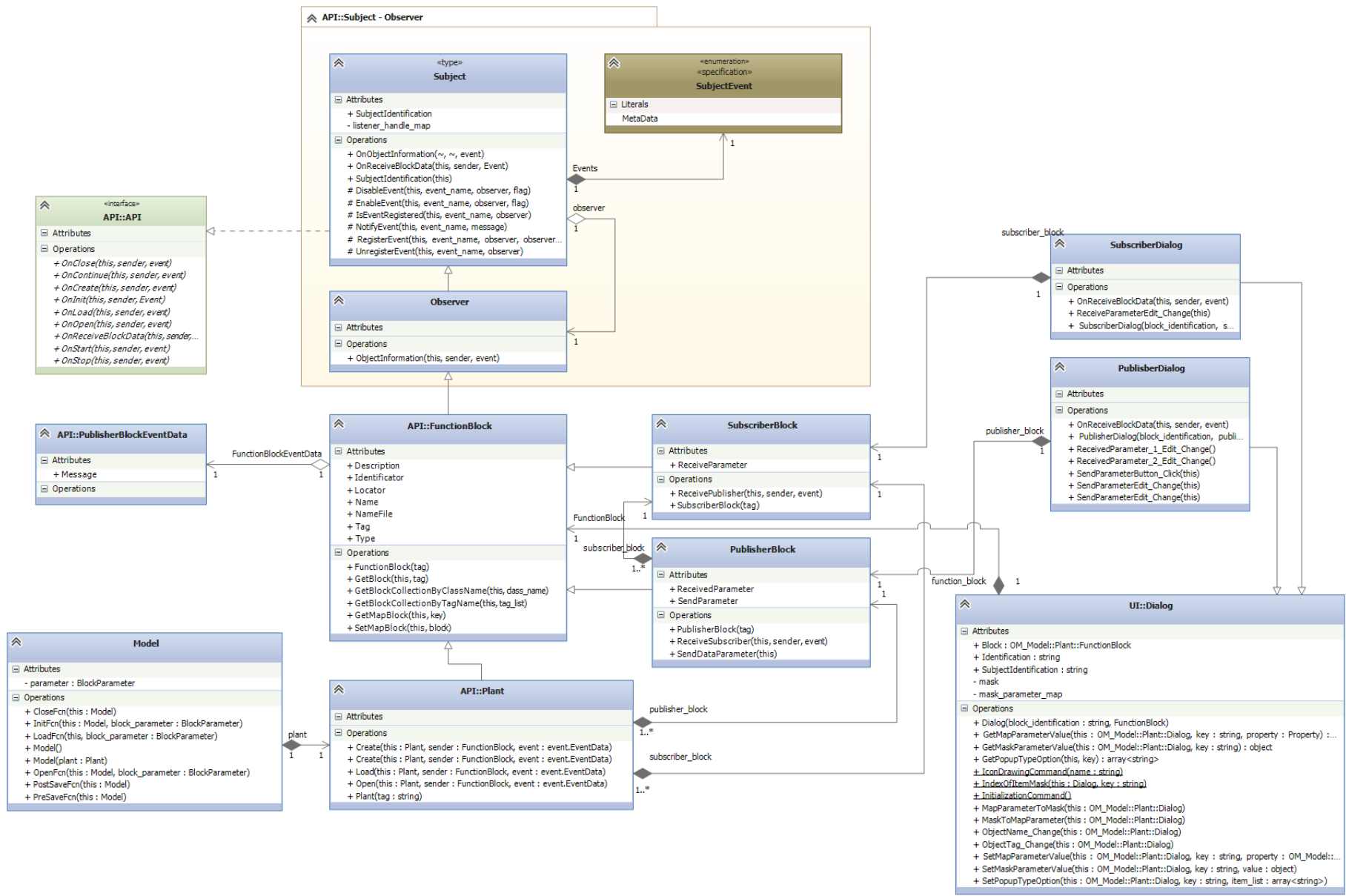


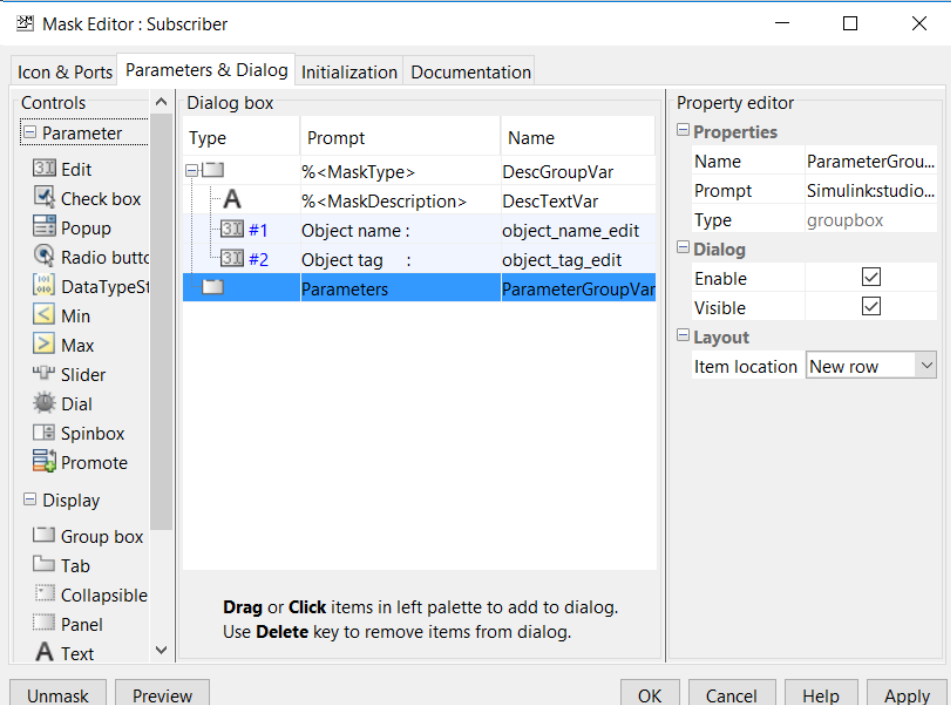
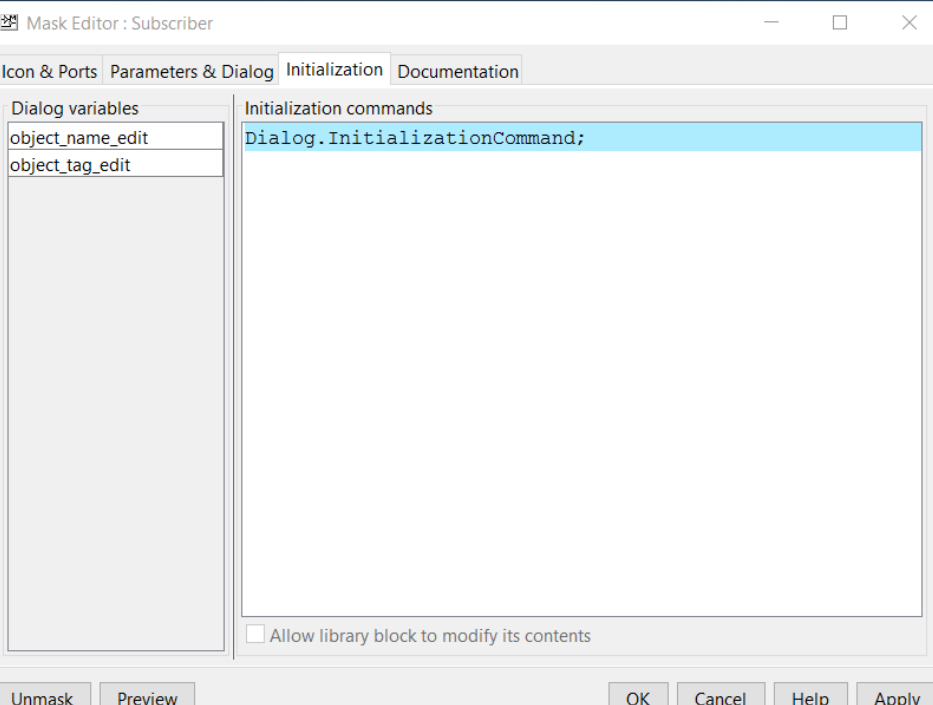
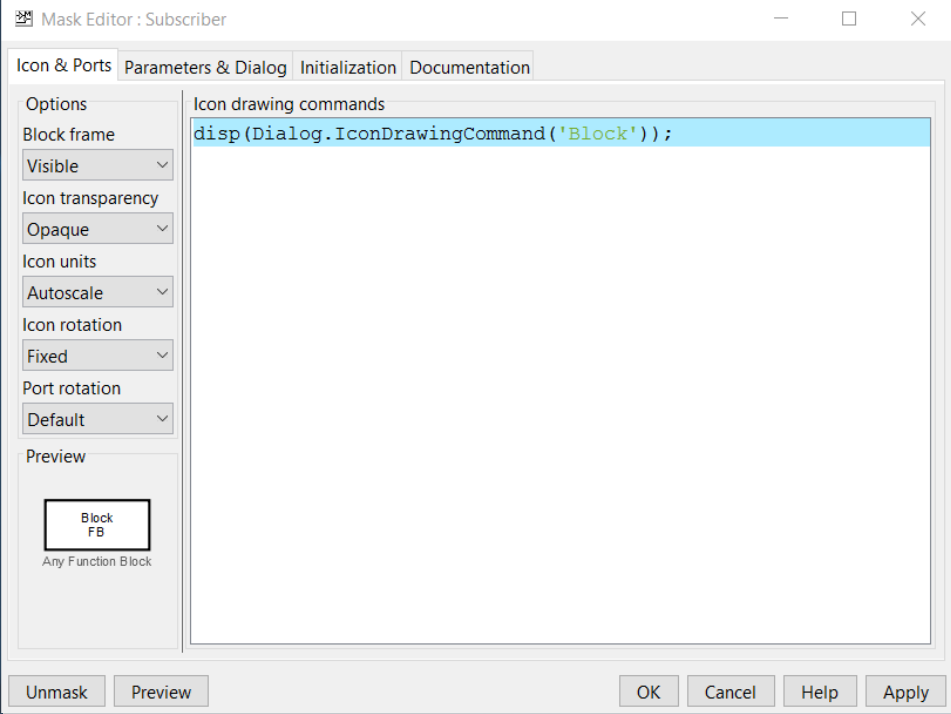
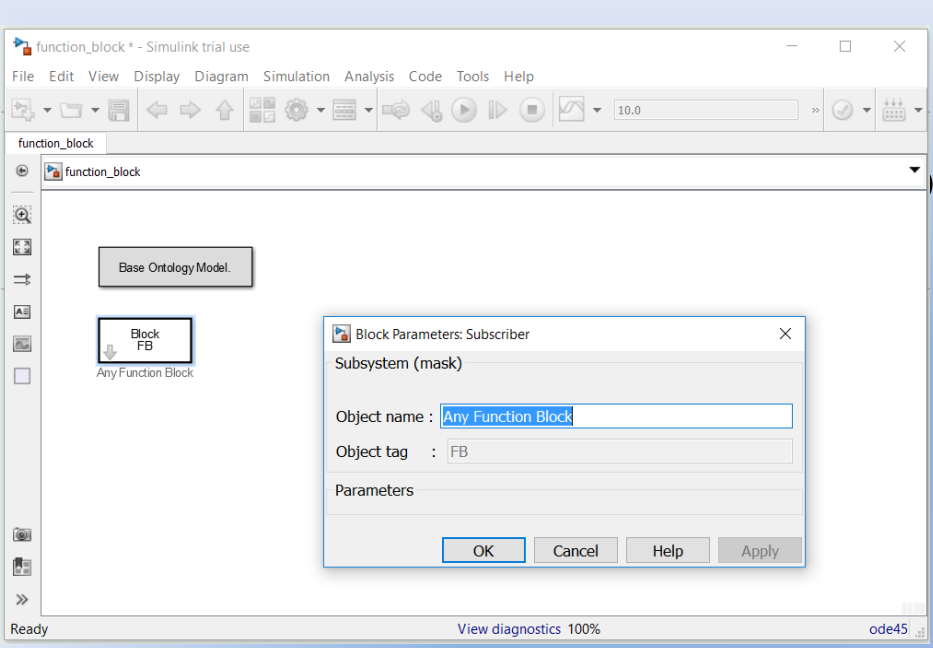
Air Oil Exchanger



Fuel Oil Exchanger







Model Properties: publisher_subscriber_channel_1

Main Callbacks History Description Data

Model callbacks

- PreLoadFcn*
- PostLoadFcn
- InitFcn*
- StartFcn
- PauseFcn
- ContinueFcn
- StopFcn
- PreSaveFcn*
- PostSaveFcn*
- CloseFcn*

Model pre-load function:

```
clc
clear

model = Model('BOM');
```

OK Cancel Help Apply

publisher_subscriber_channel_1 - Simulink trial use

File Edit View Display Diagram Simulation Analysis Code Tools

publisher_subscriber_channel_1

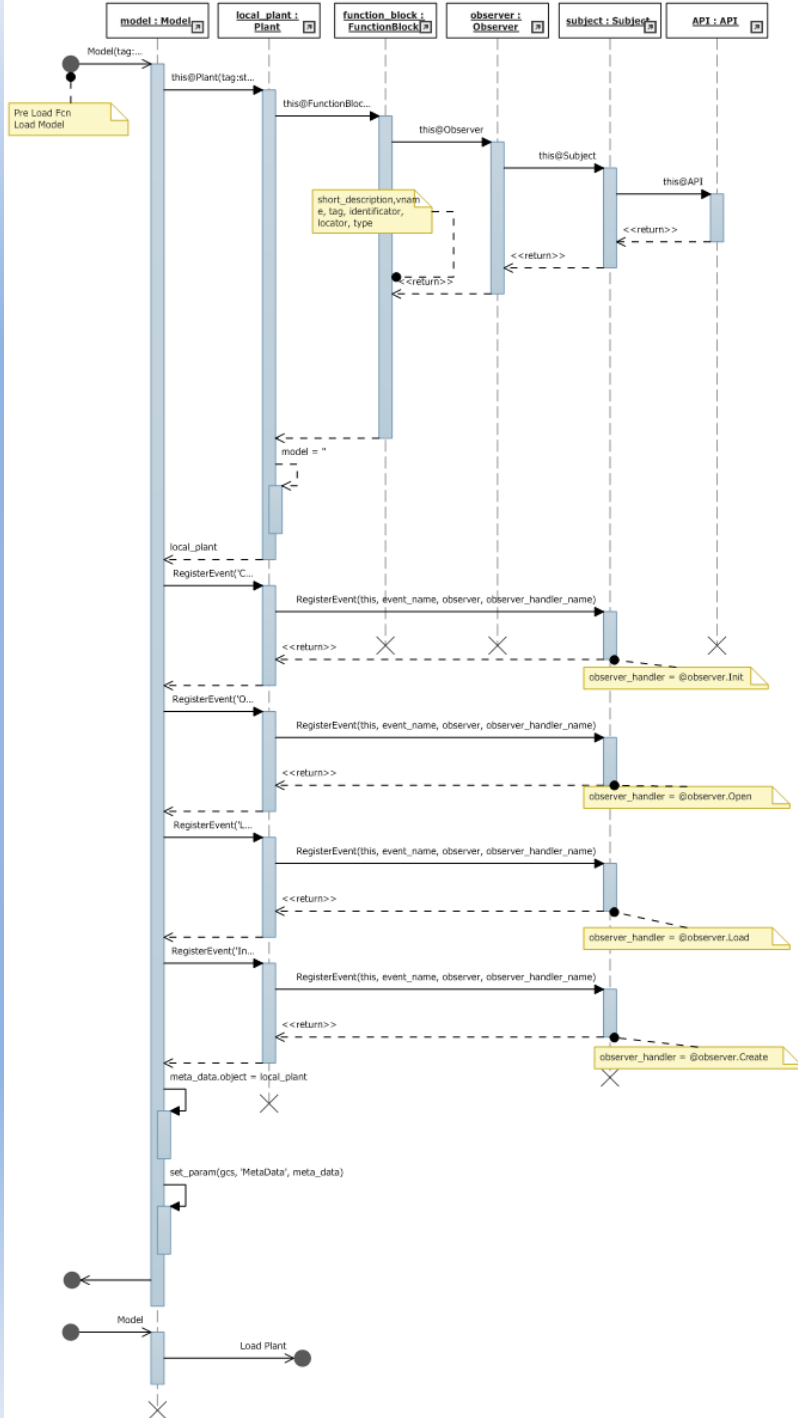
publisher_subscriber_channel_1

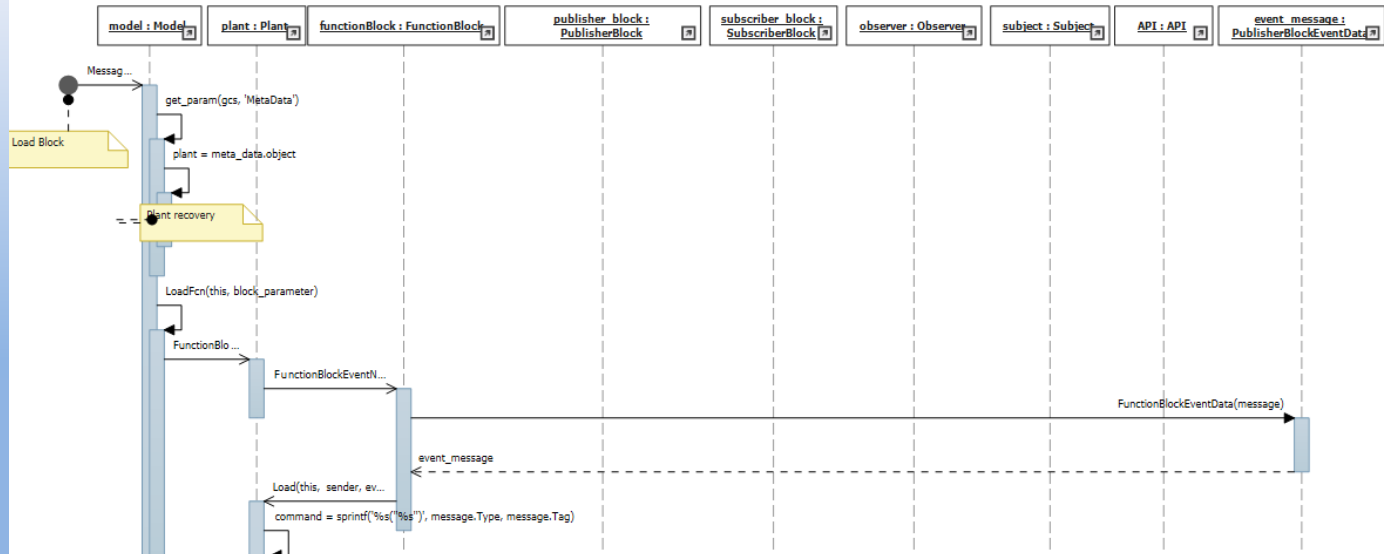
Model 1. Publisher - Subscriber.

Block FB_P
Publisher

Block FB_S1 Block FB_S2
Subscriber 1 Subscriber 2

Ready View diagnostics 100% ode45





Block Properties: Publisher

General | Block Annotation | Callbacks

Usage

To create or edit a callback function for this block, select it in the callback list (below, left). Then enter MATLAB code that implements the function in the content pane (below, right). The callback name's suffix indicates its status: *(has saved content).

Callback functions list:

- ClipboardFcn
- CloseFcn
- ContinueFcn
- CopyFcn
- DeleteChildFcn
- DeleteFcn
- DestroyFcn
- InitFcn*
- LoadFcn***
- ModelCloseFcn
- MoveFcn
- NameChangeFcn
- OpenFcn
- ParentCloseFcn
- PauseFcn
- PostSaveFcn
- PreCopyFcn
- PreDeleteFcn
- PreSaveFcn
- SCDConfigFcn

Content of callback function: "LoadFcn"

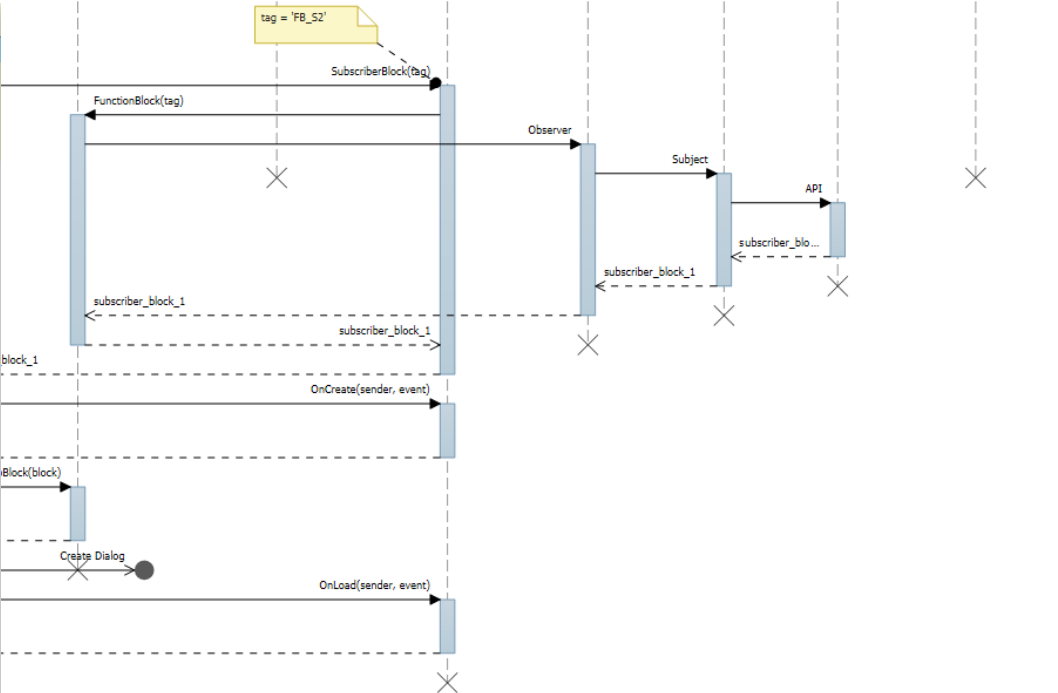
```

model = Model;
parameter = Model.BlockParameter;

parameter.EventName = 'LoadBlock';
parameter.Type = 'PublisherBlock_1';
parameter.Tag = 'FB_P';
parameter.Name = 'Publisher';
parameter.DialogType = 'PublisherDialog_1';

model.LoadFcn(parameter);
  
```

Buttons: OK, Cancel, Help, Apply



PU

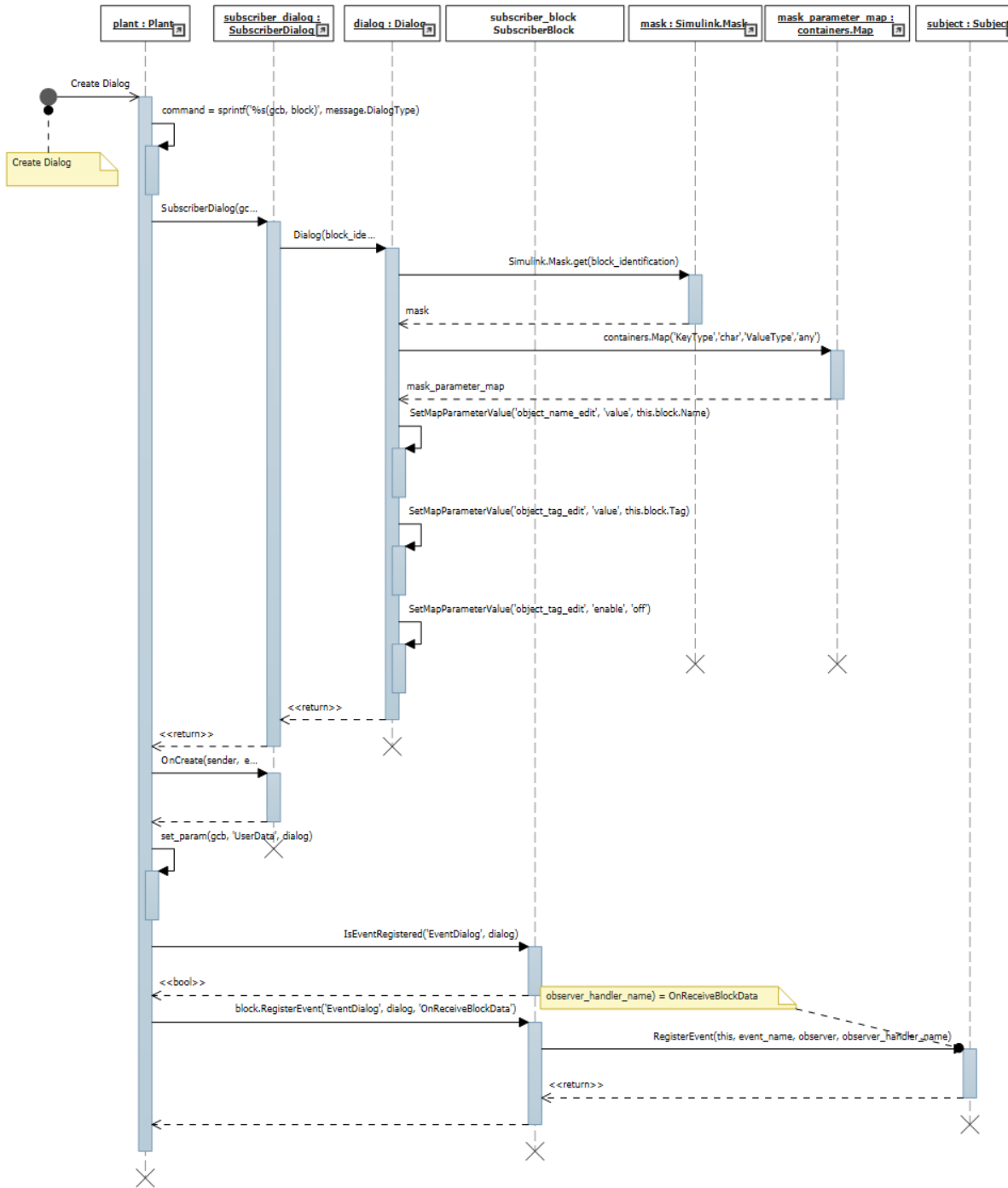
Function Block Parameters: Model

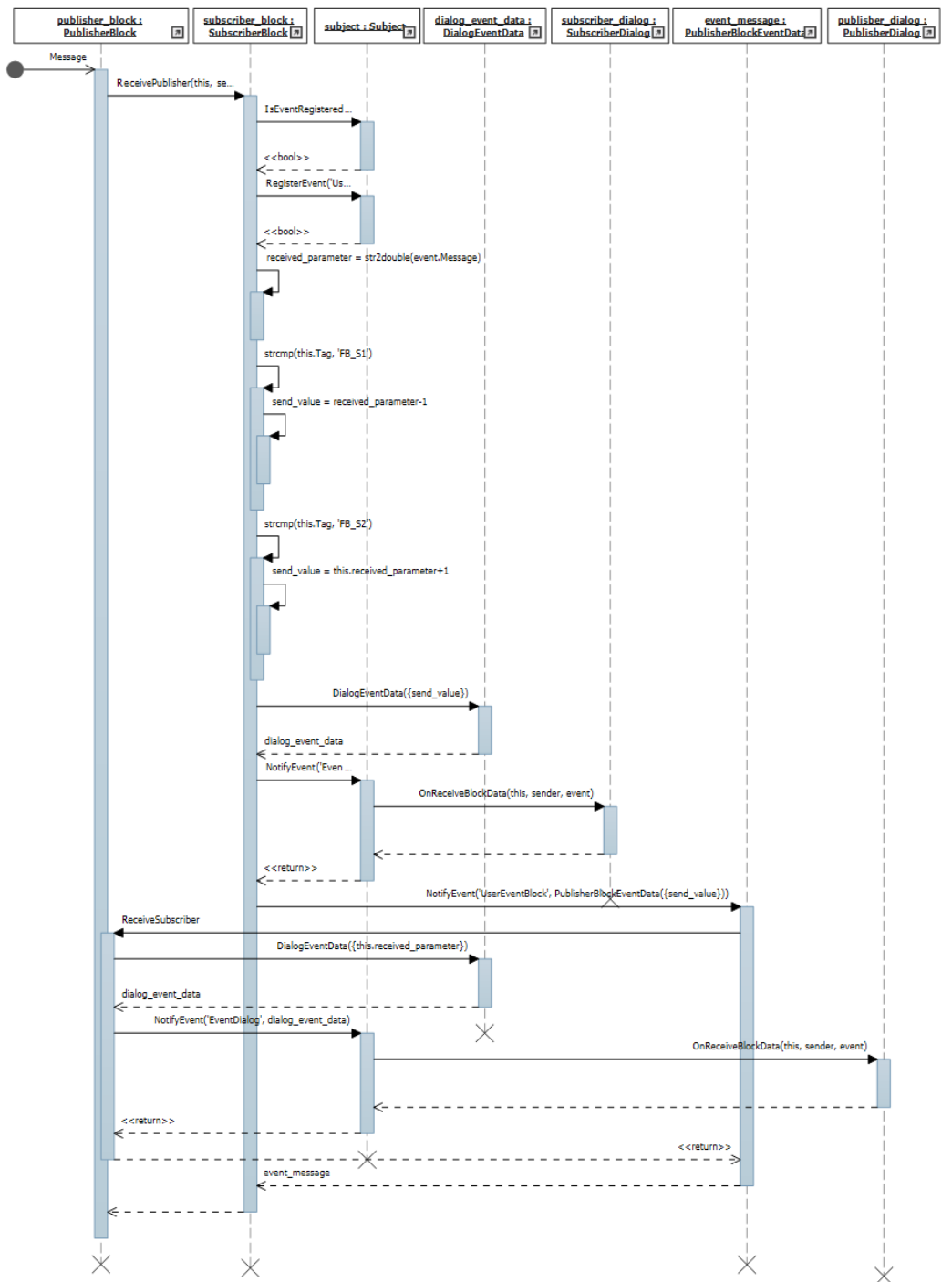
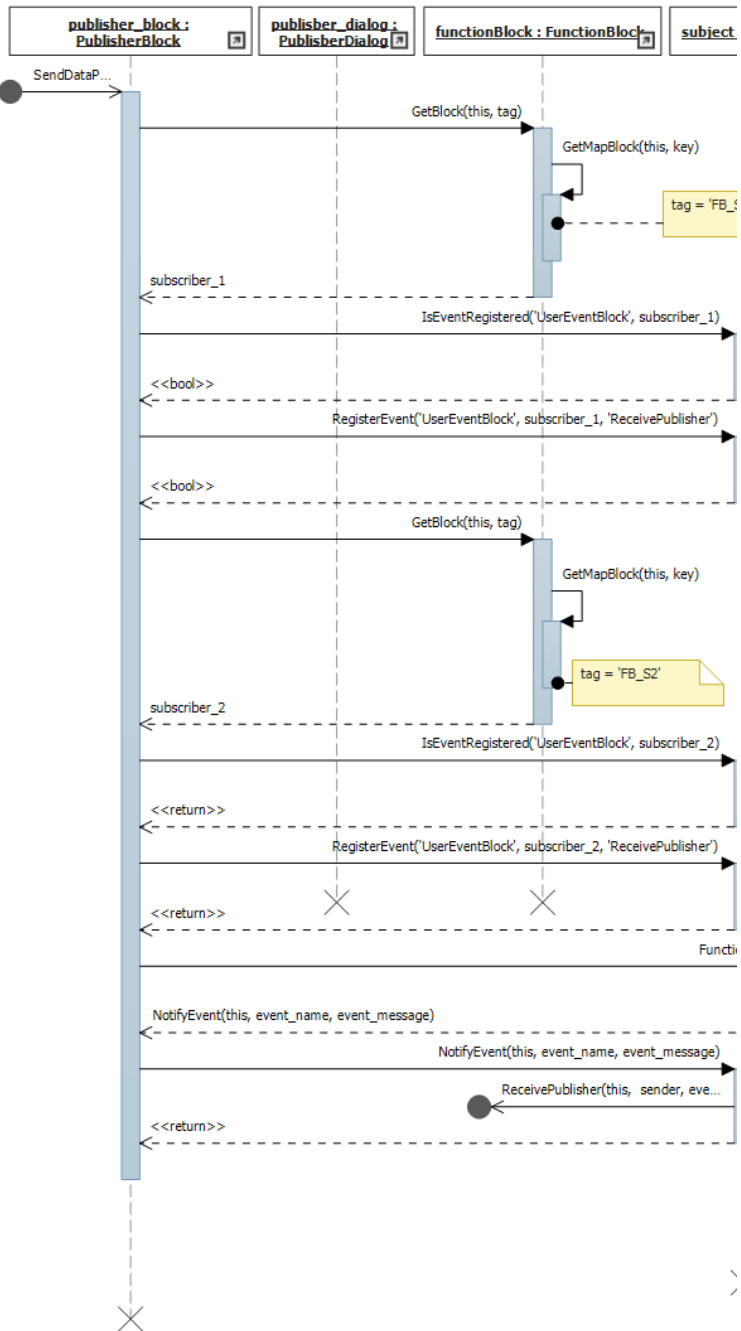
Subsystem (mask)

Object name :

Object tag :

Parameters





Block Parameters: Publisher

Subsystem (mask)

Object name :

Object tag :

Parameters

Parameter :

Receive Parameter 1 :

Receive Parameter 2 :

Receive Parameter 3 :

Block Parameters: Publisher1

Subsystem (mask)

Object name :

Object tag :

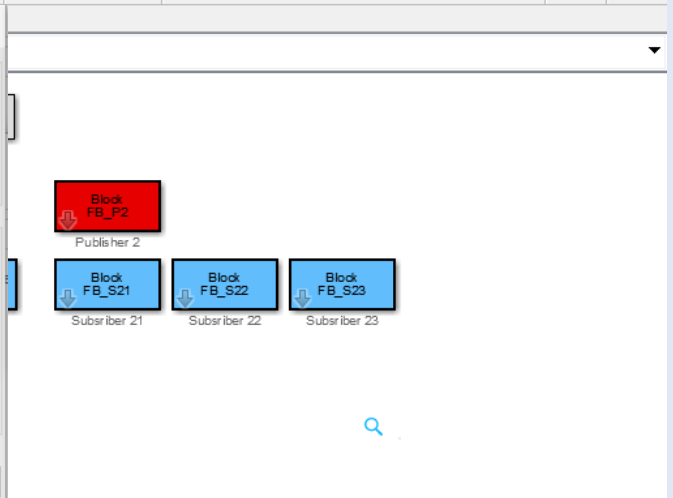
Parameters

Parameter :

Receive Parameter 1 :

Receive Parameter 2 :

Receive Parameter 3 :



Block Parameters: Subscriber 1

Subsystem (mask)

Object name :

Object tag :

Parameters

Receive Parameter :

Block Parameters: Subscriber 3

Subsystem (mask)

Object name :

Object tag :

Parameters

Receive Parameter :

View diagnostics 100% ode45

Block Parameters: Subscriber 2

Subsystem (mask)

Object name :

Object tag :

Parameters

Receive Parameter :

Block Parameters: Subscriber 5

Subsystem (mask)

Object name :

Object tag :

Parameters

Receive Parameter :

Block Parameters: Subscriber 4

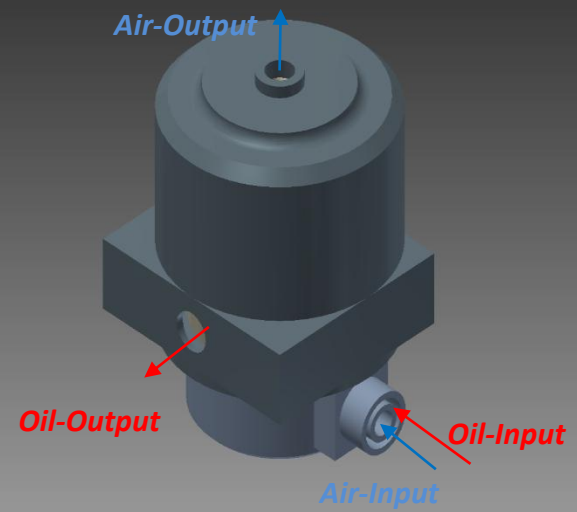
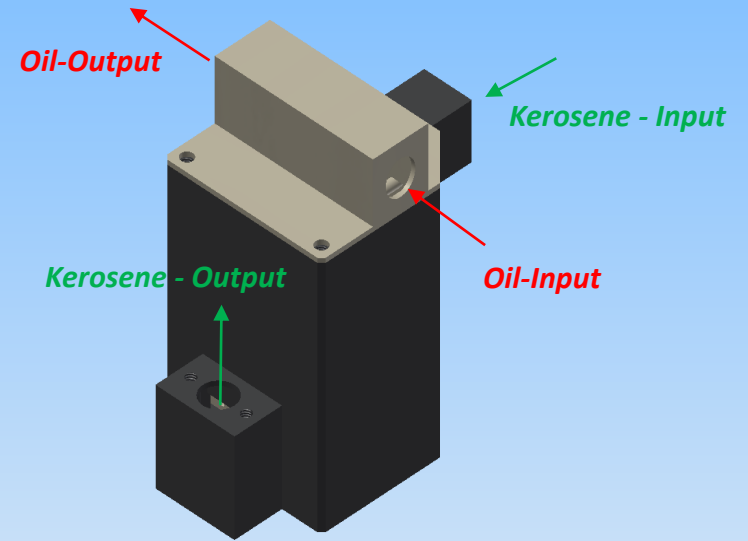
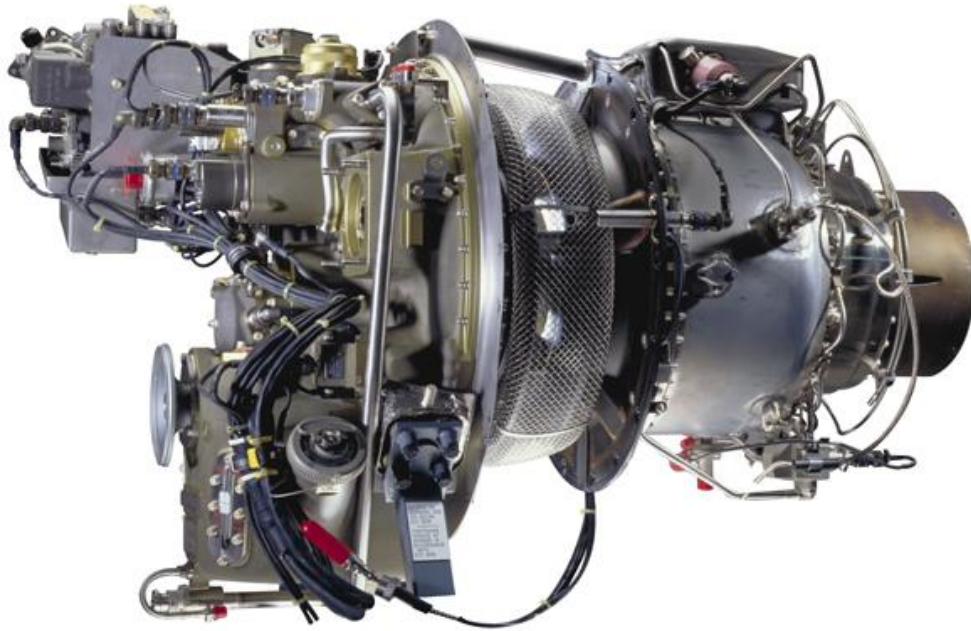
Subsystem (mask)

Object name :

Object tag :

Parameters

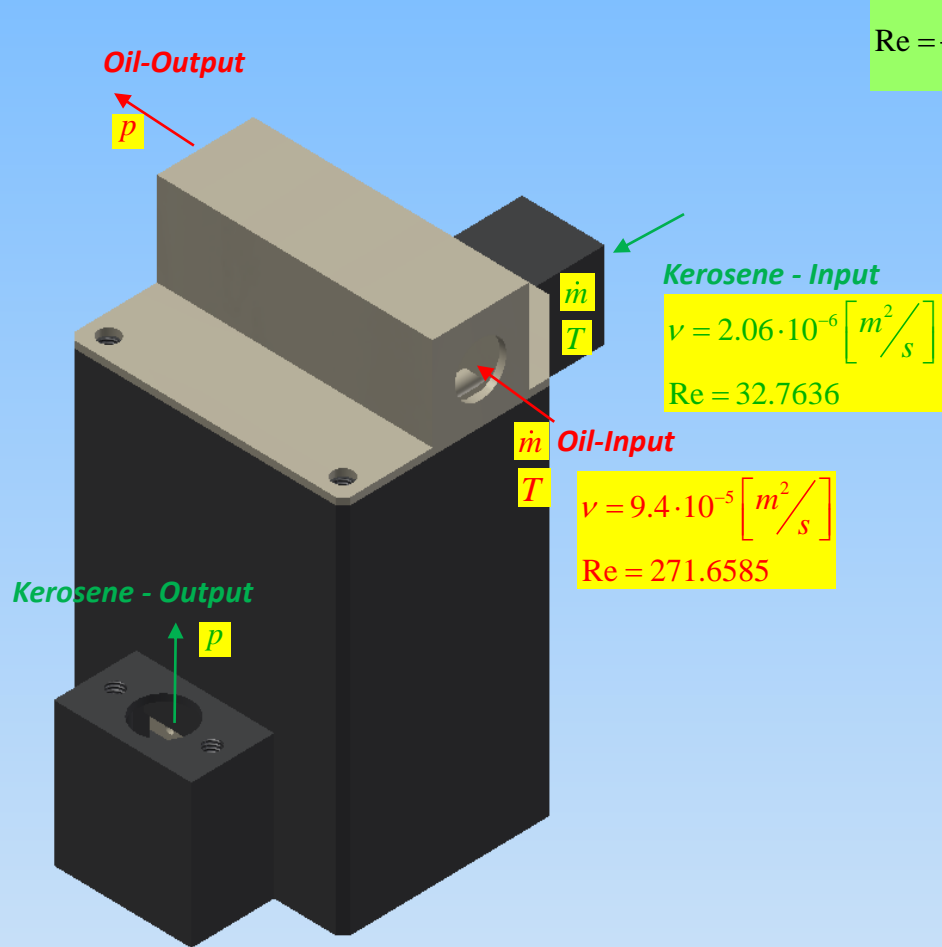
Receive Parameter :



Kerosene		
\dot{m}	310.0	kg/h
T	10.0	$^{\circ}\text{C}$
p	150.0	kPa

Oil		
\dot{m}	50.0	kg/h
T	145.0	$^{\circ}\text{C}$
p	210.0	kPa

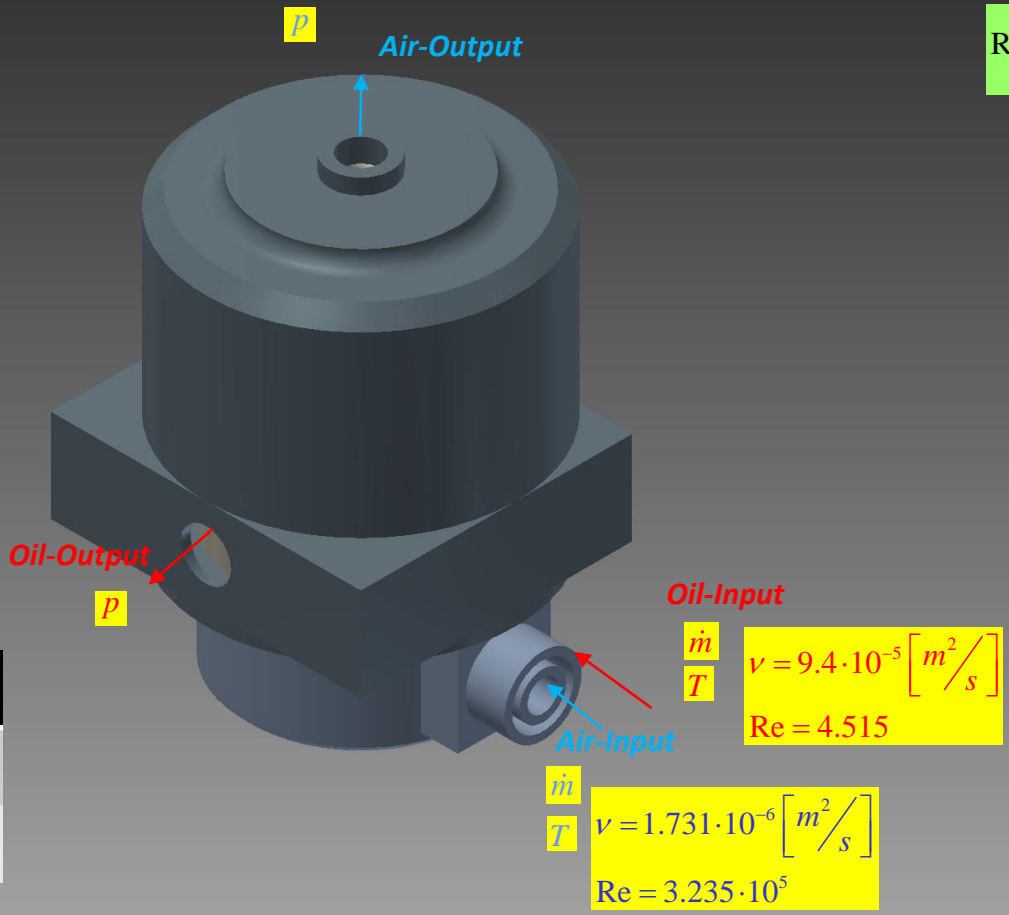
Surroundings		
T	20.0	$^{\circ}\text{C}$
p	101.325	kPa



Air		
u	280.0	m/s
T	40.0	$^{\circ}C$
p	101.325	kPa

Oil		
\dot{V}	72.0	dm^3/h
T	145.0	$^{\circ}C$
p	202.65	kPa

Surroundings		
T	20.0	$^{\circ}C$
p	101.325	kPa



$$Re = \frac{F_i}{F_v} = \frac{u \cdot l}{\nu}$$



Fourier-Kirchhoff :

$$c_p \cdot \rho \cdot \frac{\partial T}{\partial t} + c_p \cdot \rho \cdot u \cdot \nabla T + \nabla q = Q$$

$$q = -k \cdot \nabla T$$

$$\rightarrow u \left[\frac{m}{s} \right]$$

$$\rho = 7850 \left[\frac{kg}{m^3} \right]$$

$$c_p = 475 \left[\frac{J}{(kg \cdot K)} \right]$$

$$k = 44.5 \left[\frac{W}{(m \cdot K)} \right]$$

$$\rho = 1089.289 - 1.002177 \cdot T^1 \left[\frac{kg}{m^3} \right]$$

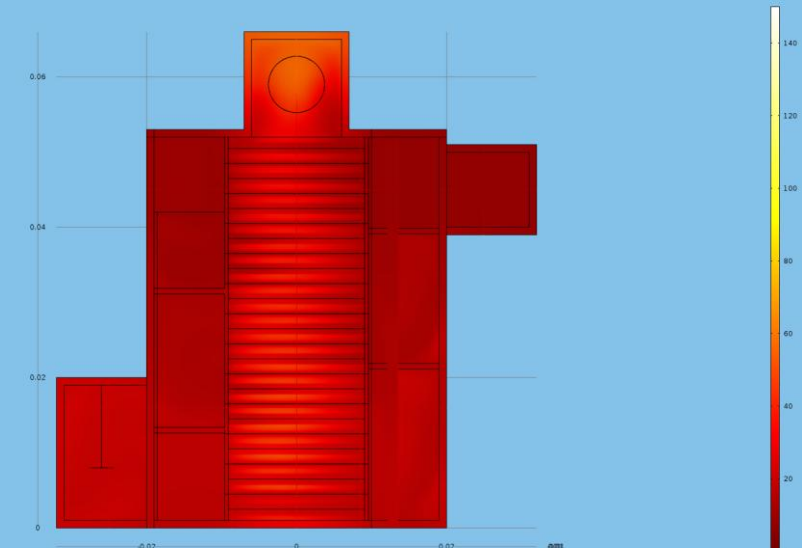
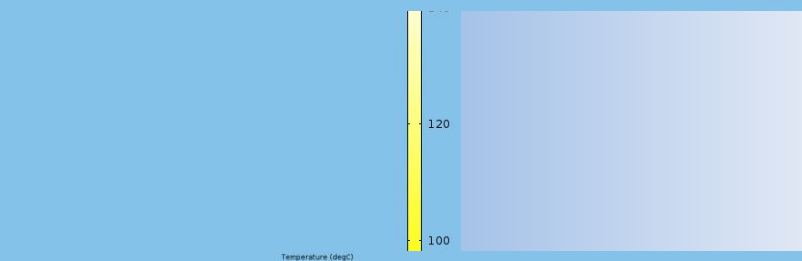
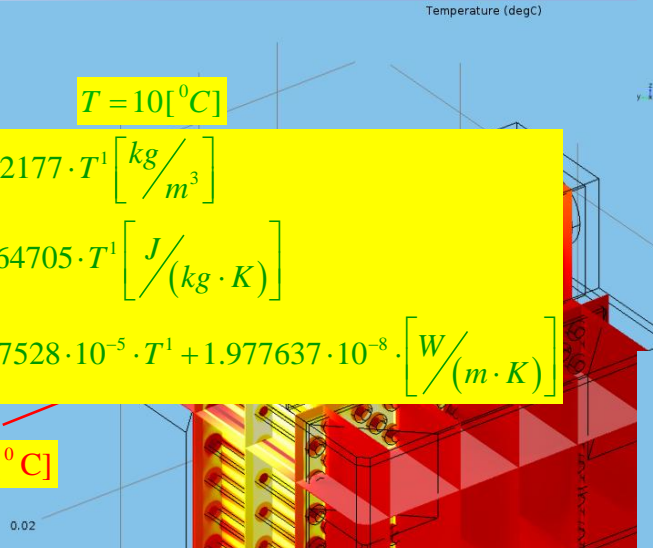
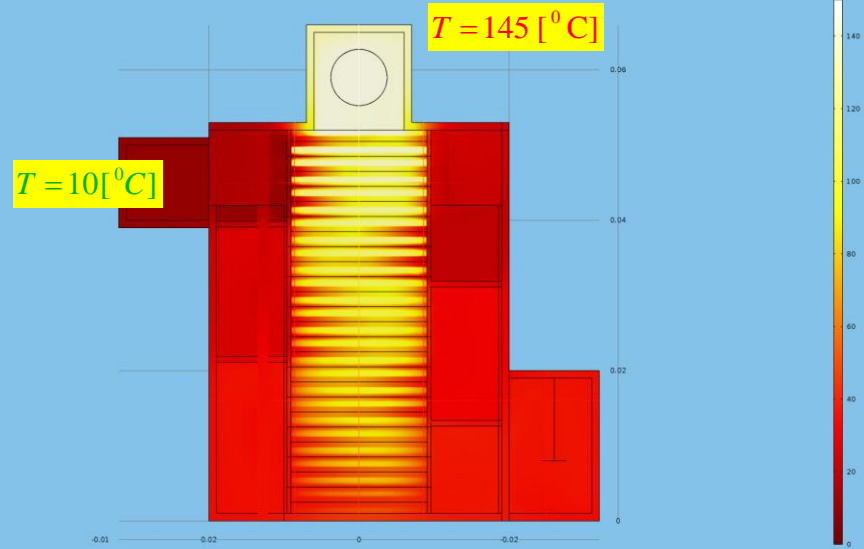
$$c_p = 854.5883 + 3.764705 \cdot T^1 \left[\frac{J}{(kg \cdot K)} \right]$$

$$k = 0.148681 - 6.487528 \cdot 10^{-5} \cdot T^1 + 1.977637 \cdot 10^{-8} \cdot T^2 \left[\frac{W}{(m \cdot K)} \right]$$

$$\rho = 1068.70404 - 0.6393421 \cdot T^1 + 7.34307359 \cdot 10^{-5} \cdot T^2 \left[\frac{kg}{m^3} \right]$$

$$c_p = 761.405625 + 3.47685606 \cdot T^1 + 0.00115530303 \cdot T^2 \left[\frac{J}{(kg \cdot K)} \right]$$

$$k = 0.192223542 - 2.0637987 \cdot 10^{-4} \cdot T^1 + 1.54220779 \cdot 10^{-7} \cdot T^2 \left[\frac{W}{(m \cdot K)} \right]$$



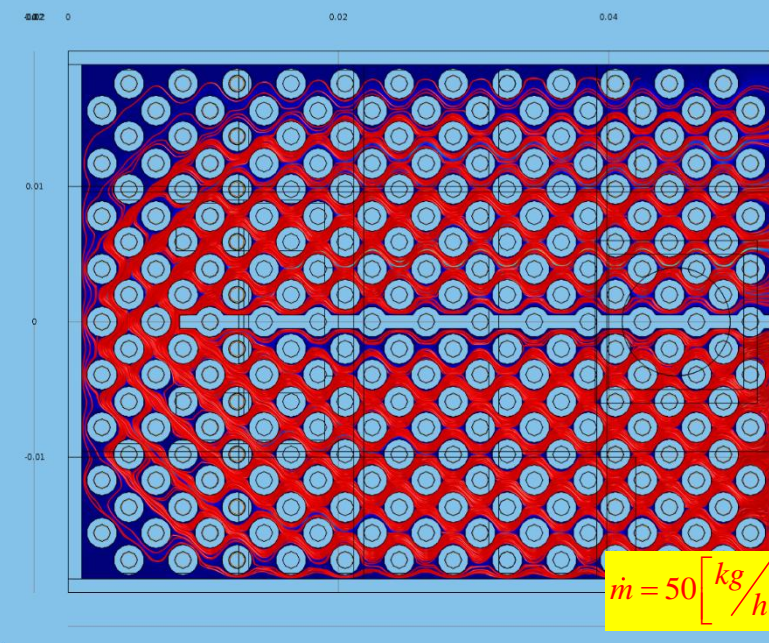
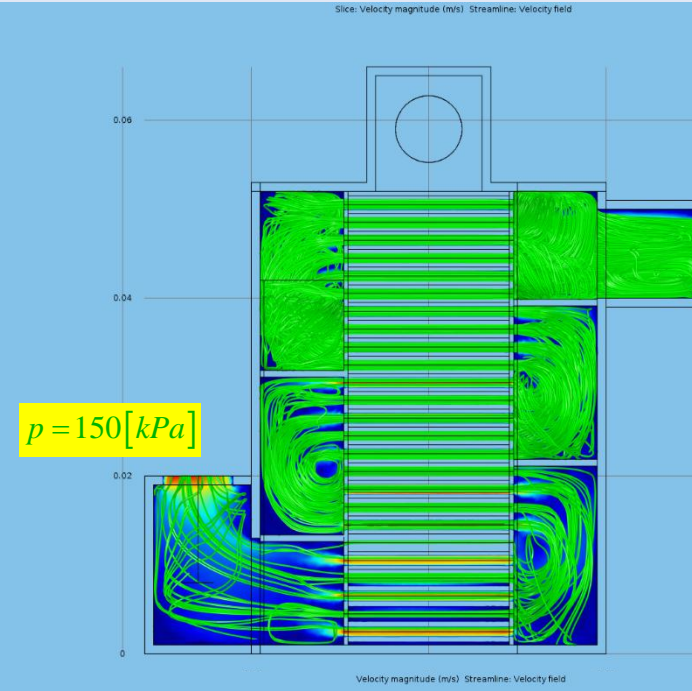
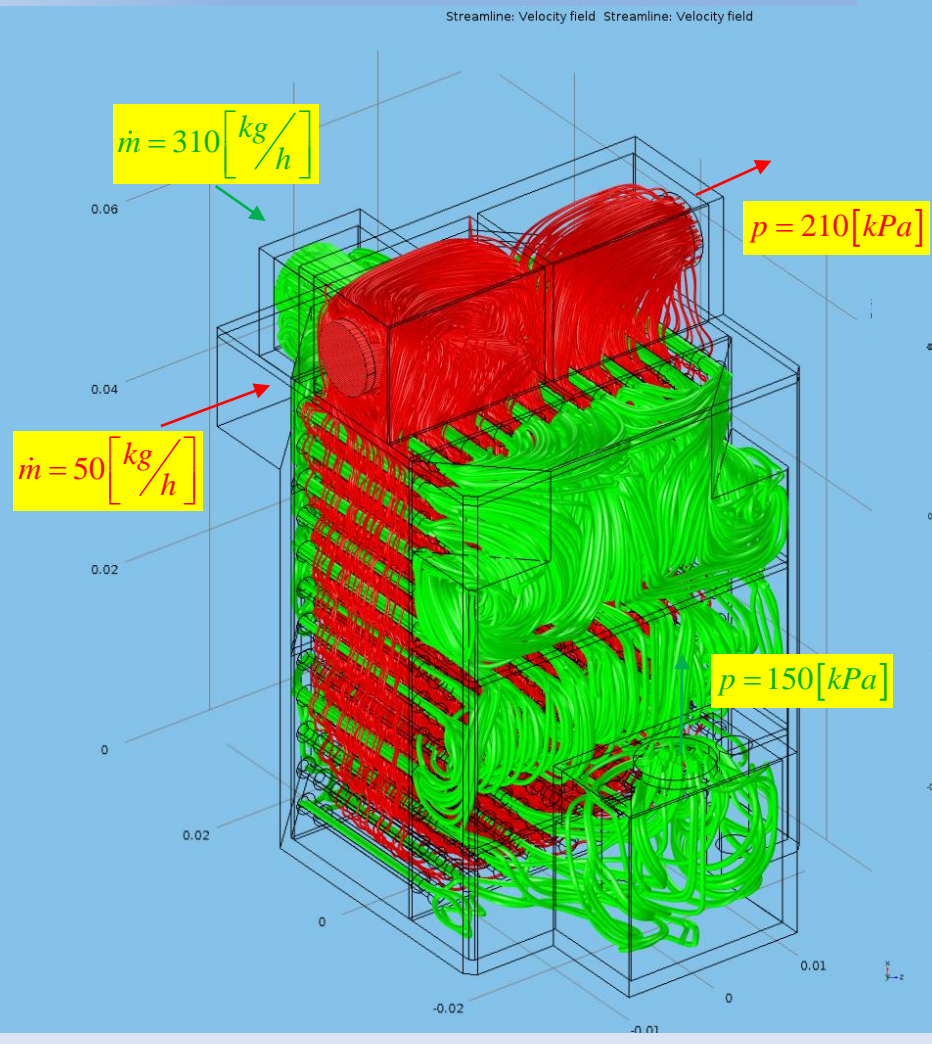
Navier Stokes :

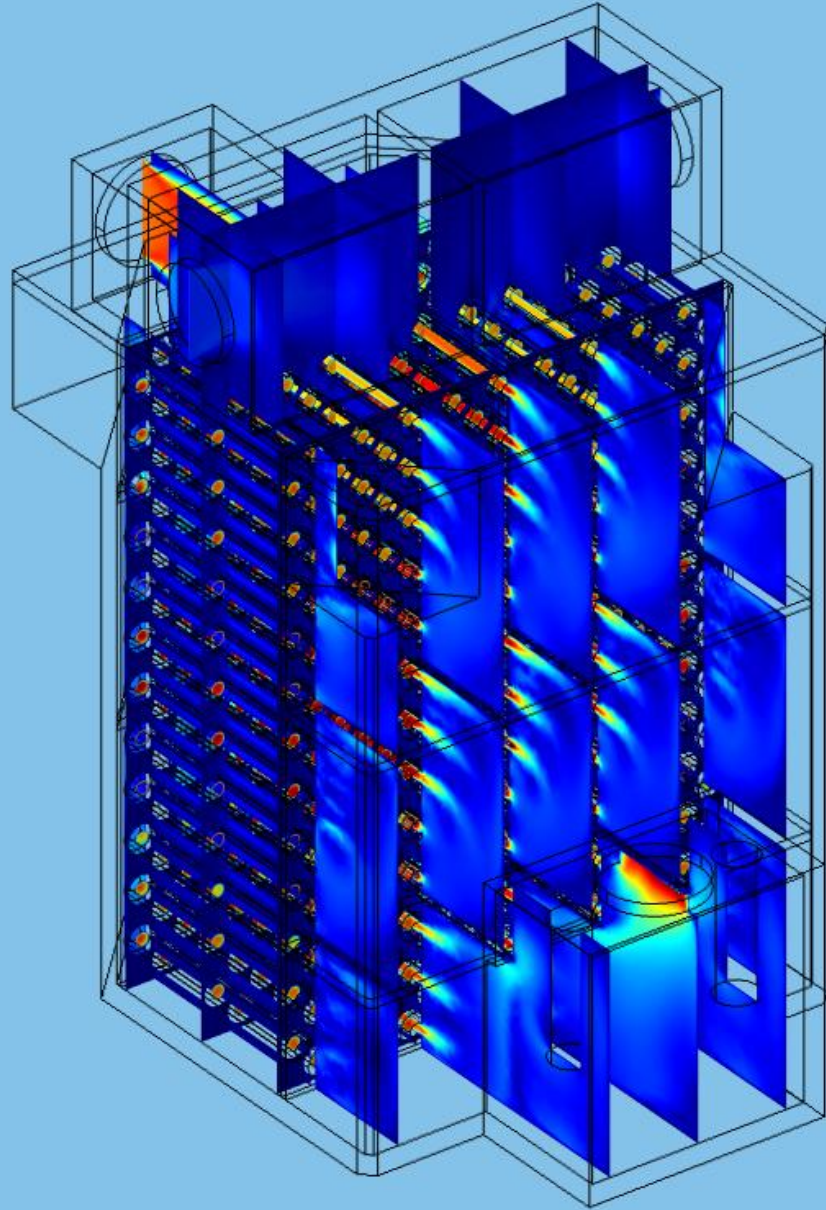
$$\rho \cdot \left(\frac{\partial u}{\partial t} + u \cdot \nabla u \right) = -\nabla p - \nabla q + Q$$

Continuity :

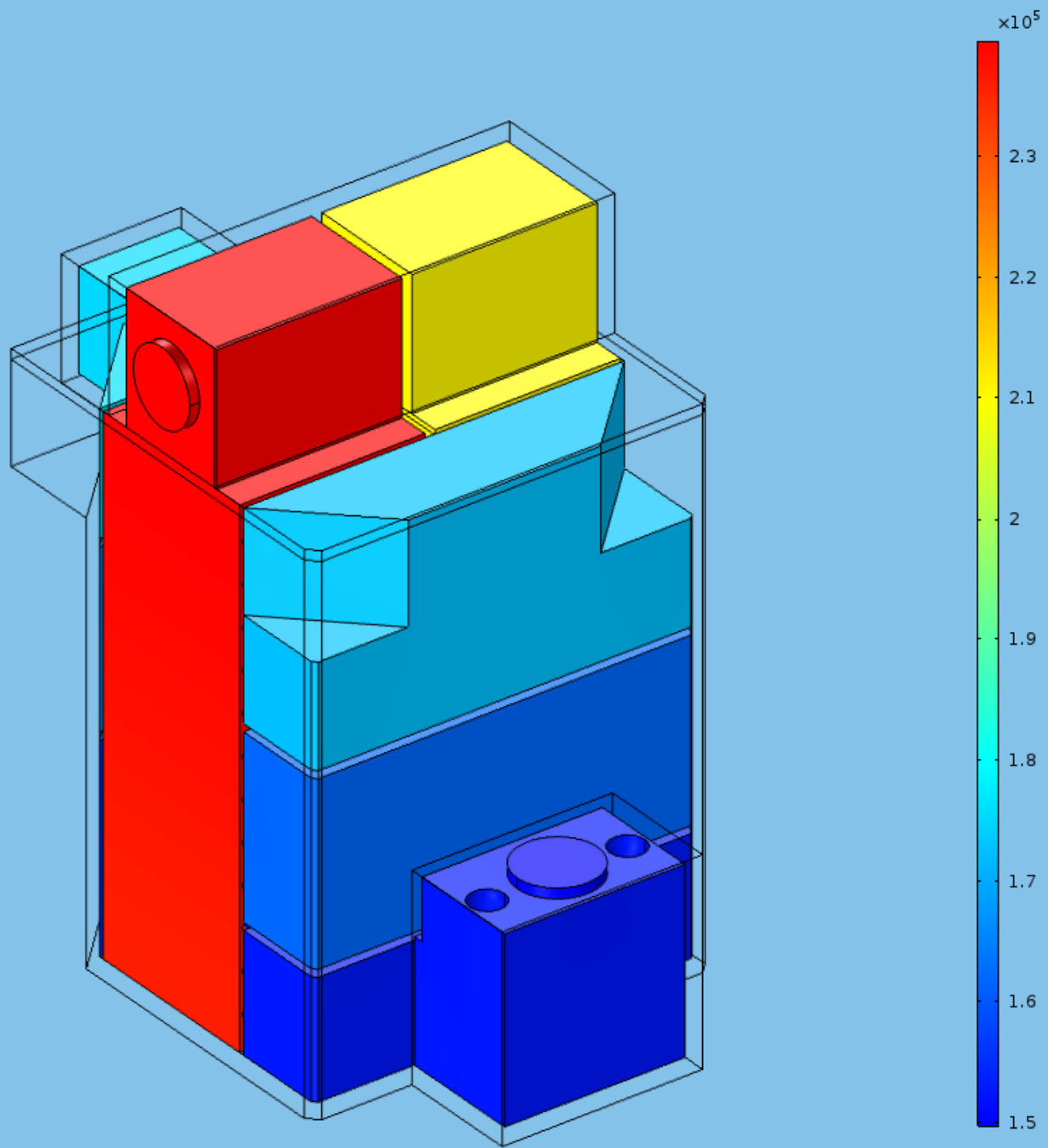
$$\frac{\partial \rho}{\partial t} + \nabla(\rho \cdot u) = 0$$

$$\rightarrow q \left[\frac{W}{m^2} \right]$$





Surface: Pressure (Pa)

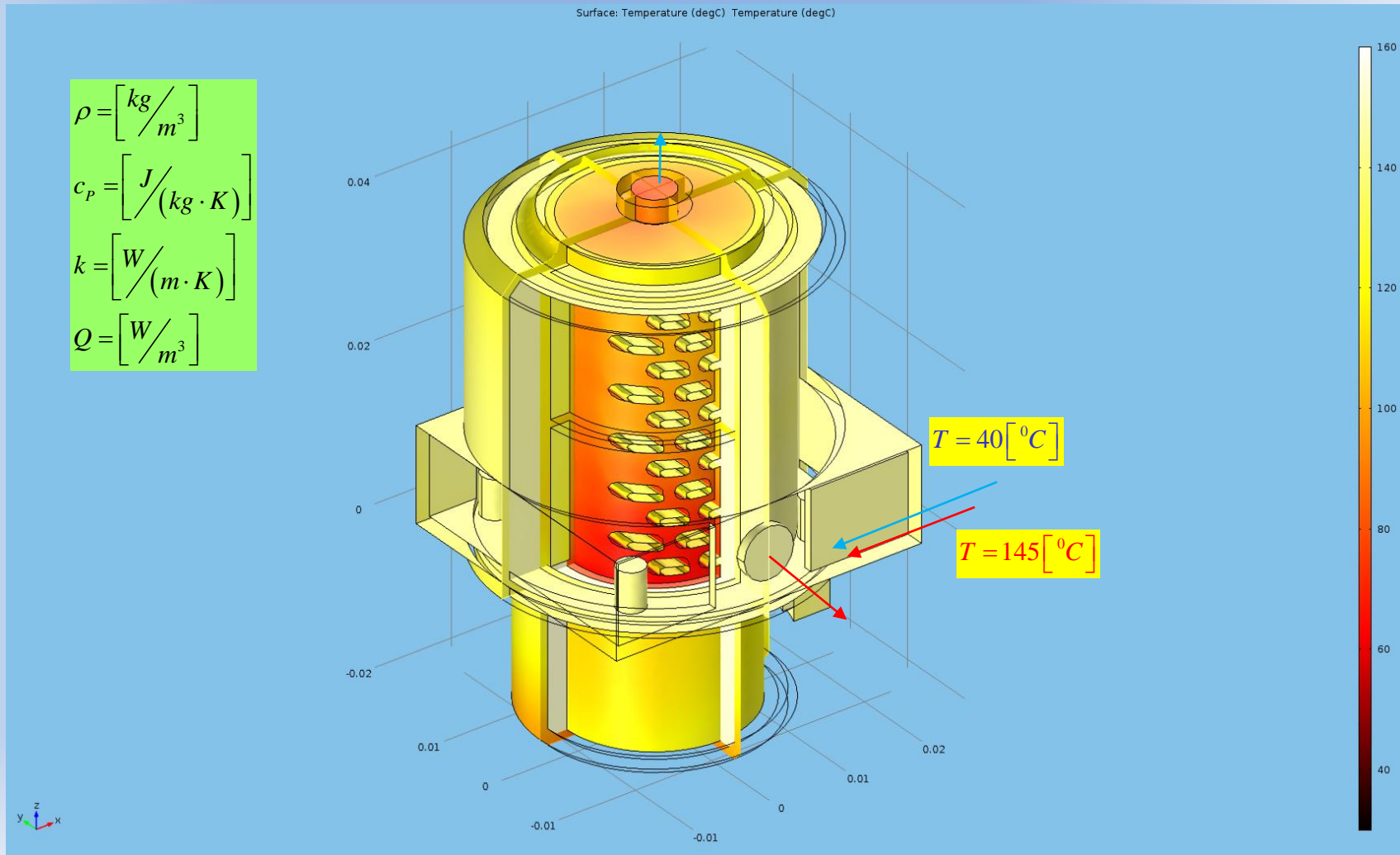


Fourier-Kirchhoff :

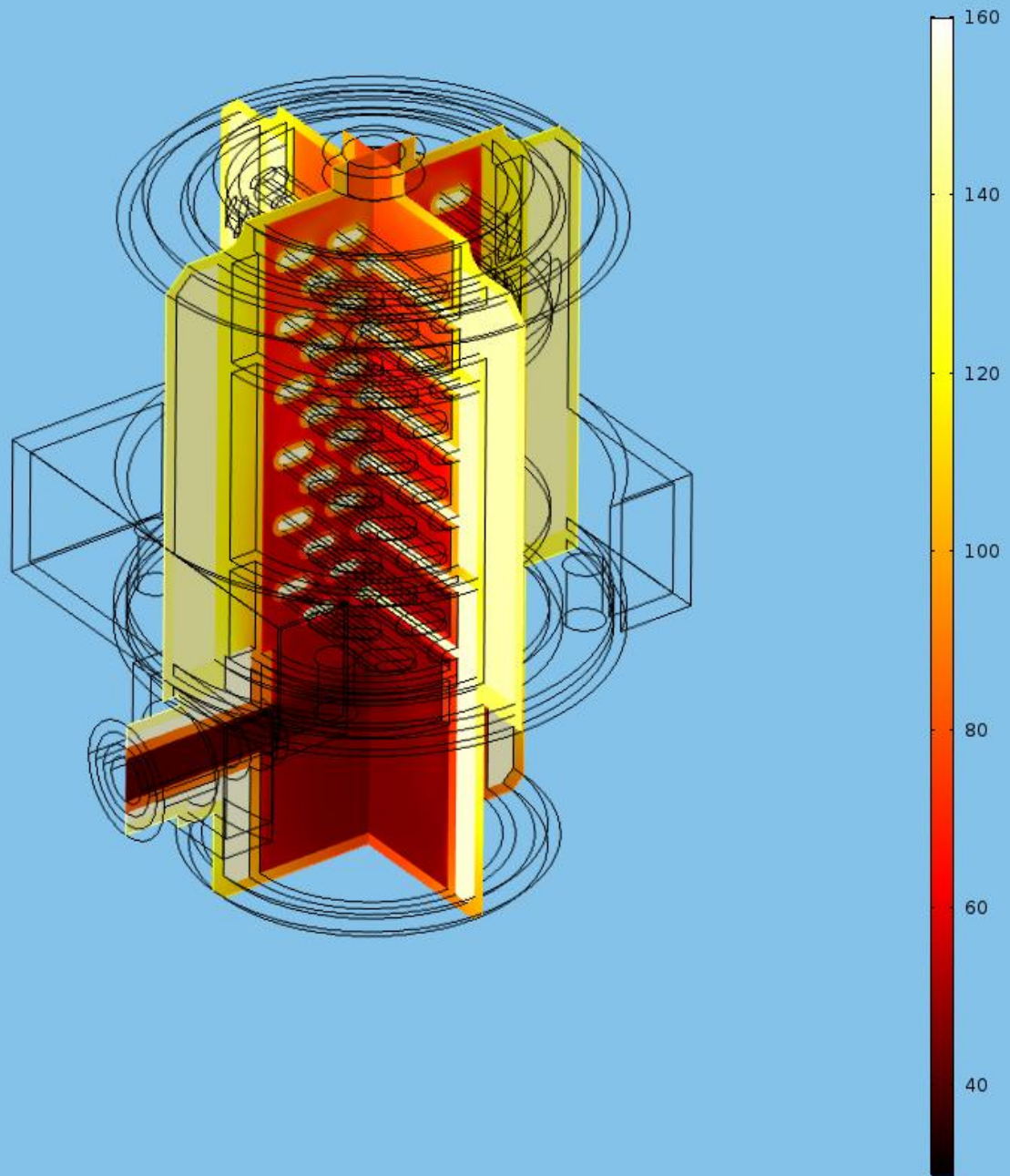
$$c_p \cdot \rho \cdot \frac{\partial T}{\partial t} + c_p \cdot \rho \cdot u \cdot \nabla T + \nabla q = Q$$
$$q = -k \cdot \nabla T$$

$$\rightarrow u \left[\frac{m}{s} \right]$$

$$\rho = \left[\frac{kg}{m^3} \right]$$
$$c_p = \left[\frac{J}{(kg \cdot K)} \right]$$
$$k = \left[\frac{W}{(m \cdot K)} \right]$$
$$Q = \left[\frac{W}{m^3} \right]$$



Temperature (degC)



k-ε model:

$$\rho \cdot \frac{\partial k}{\partial t} + \rho \cdot u \cdot \nabla k = \nabla \cdot \left(\left(\mu + \frac{\mu_T}{\sigma_k} \right) \nabla k \right) + P_k - \rho \cdot \varepsilon$$

$$P_k = \mu_T \cdot (\nabla u + \nabla^T u) - \frac{2}{3} \cdot (\nabla u)^2 - \frac{2}{3} \cdot \rho \cdot k \cdot V \cdot u$$

$$\mu_T = \rho \cdot C_\mu \cdot \frac{k^2}{\varepsilon}$$

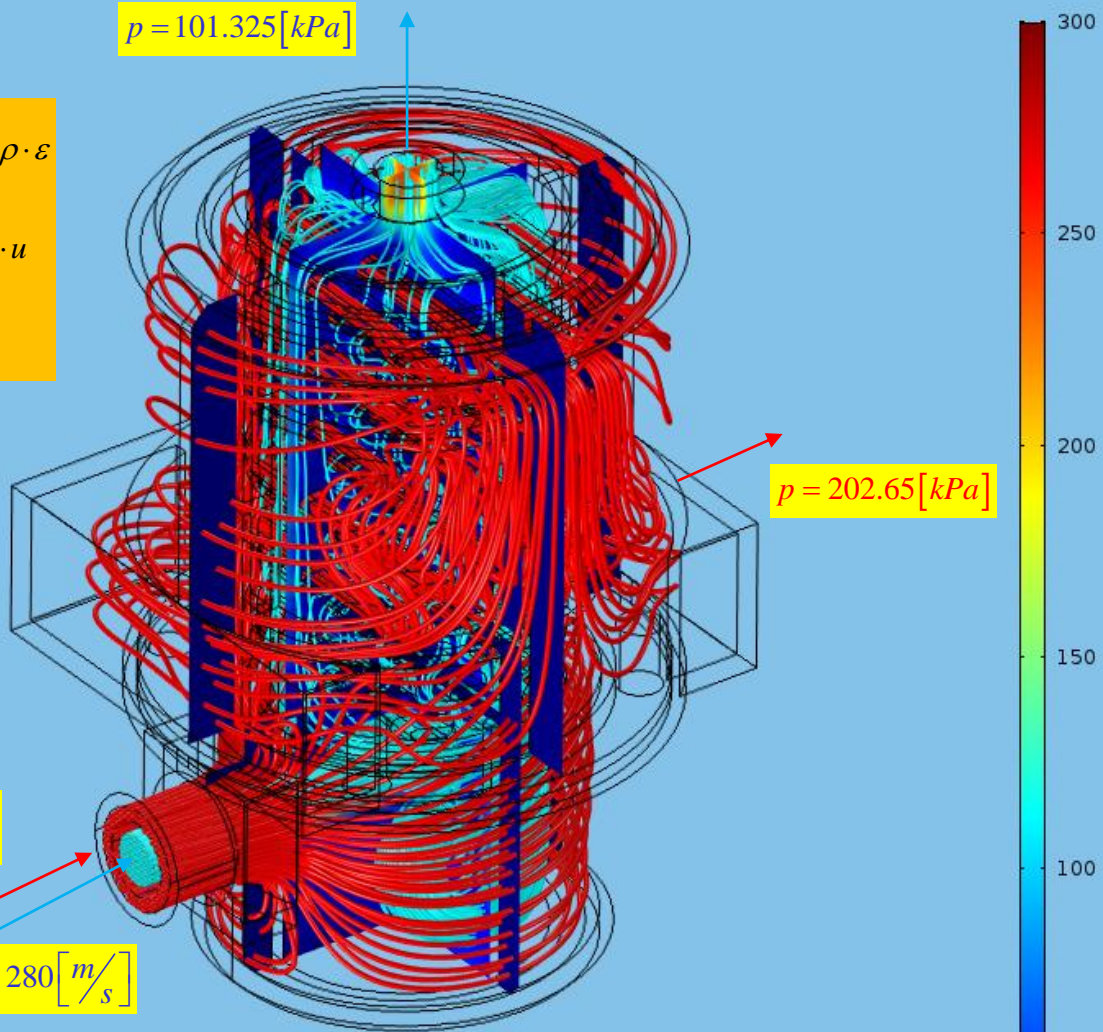
- $C_\mu = 0.09$
- $\sigma_k = 1.0$
- $\sigma_\varepsilon = 1.3$
- $C_{\varepsilon 1} = 1.44$
- $C_{\varepsilon 2} = 1.92$

$$\dot{V} = 72 \left[\frac{dm^3}{h} \right]$$

$$u = 280 \left[\frac{m}{s} \right]$$

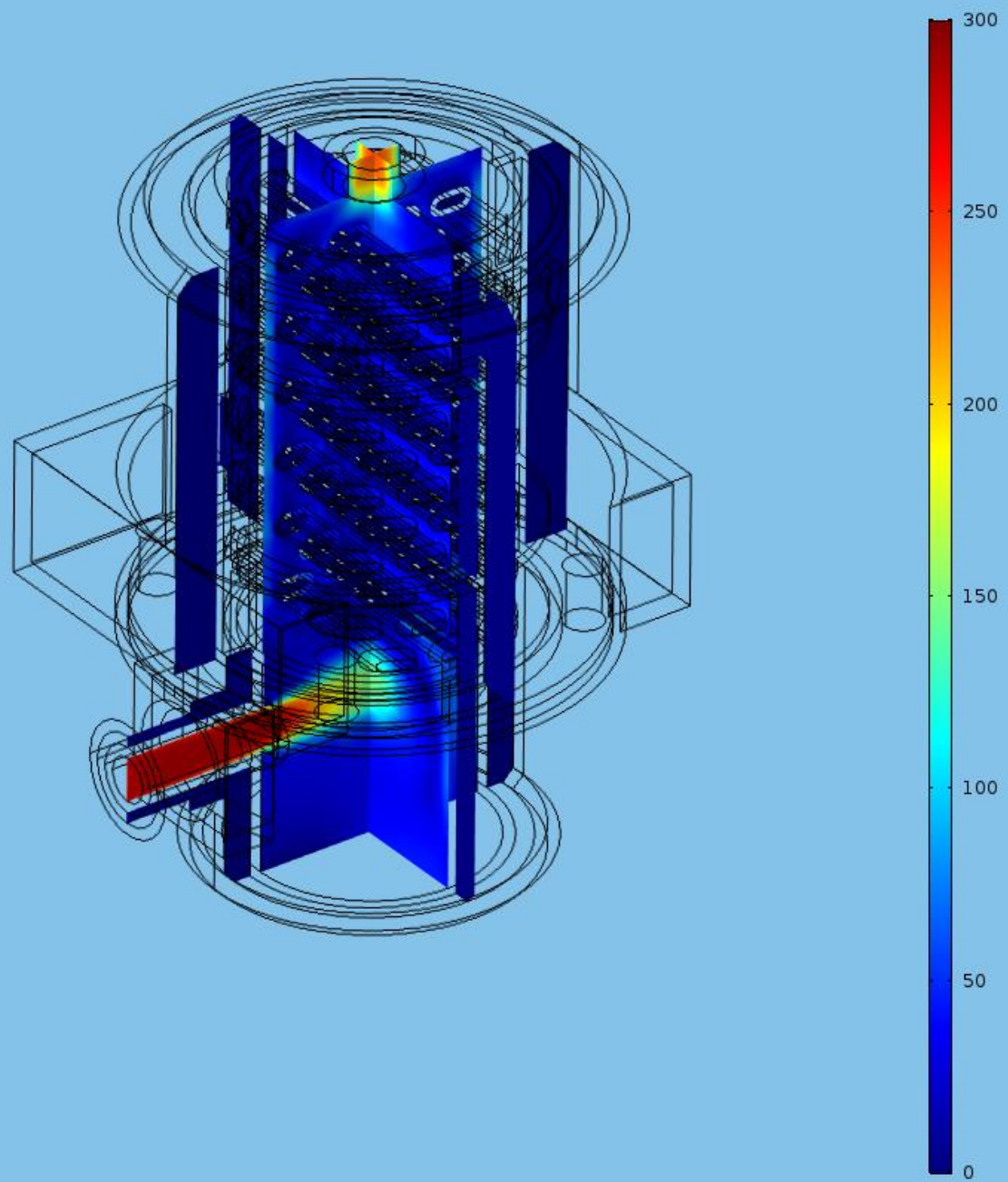
$$p = 101.325 \left[kPa \right]$$

$$p = 202.65 \left[kPa \right]$$

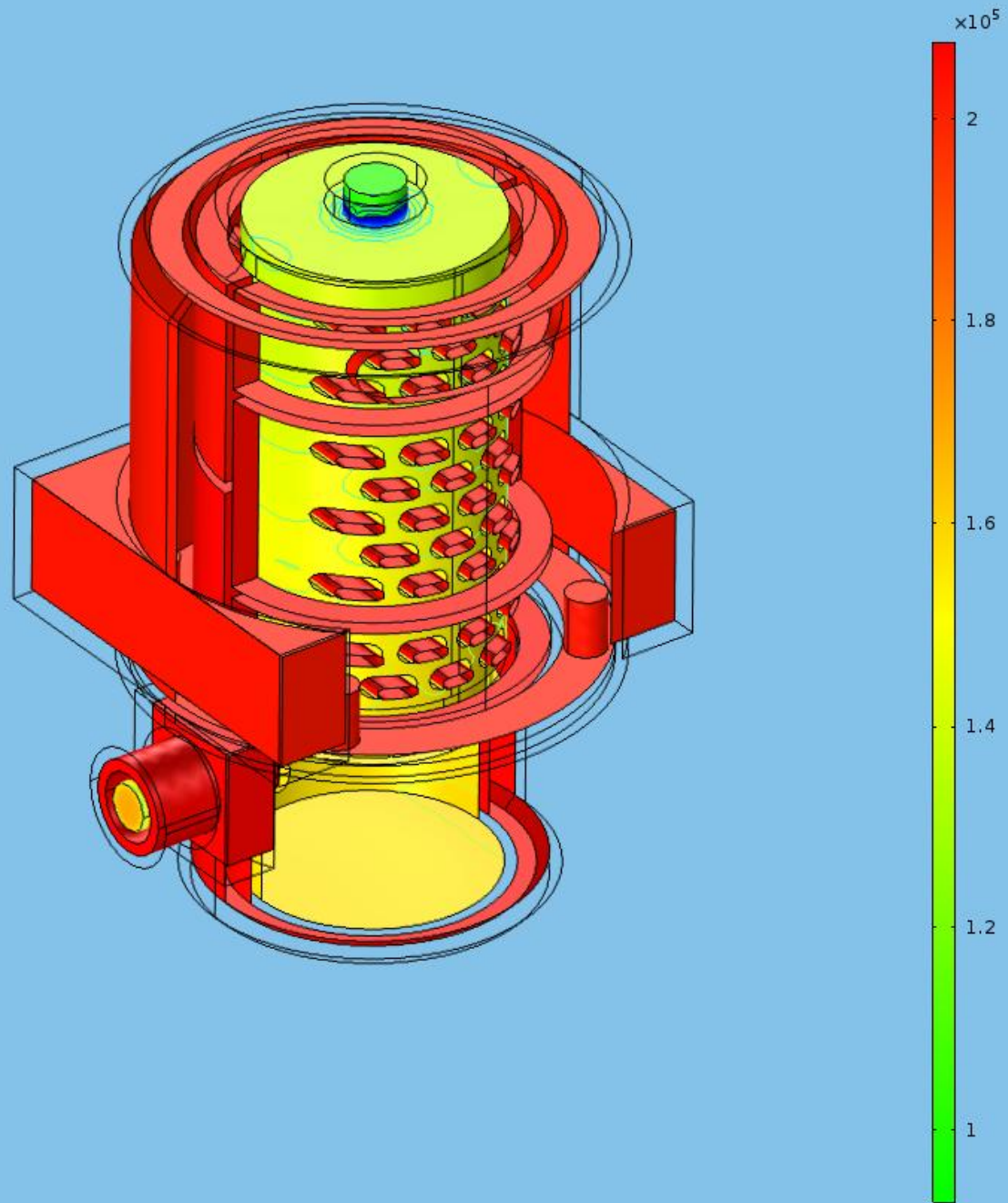


$$\rho \cdot \frac{\partial \varepsilon}{\partial t} + \rho \cdot u \cdot \nabla \varepsilon = \nabla \cdot \left(\left(\mu + \frac{\mu_T}{\sigma_\varepsilon} \right) \cdot \nabla \varepsilon \right) + C_{\varepsilon 1} \cdot \frac{\varepsilon}{k} \cdot P_k - C_{\varepsilon 2} \cdot \rho \cdot \frac{\varepsilon^2}{k} \cdot P_k$$

Slice: Velocity magnitude (m/s) Slice: Velocity magnitude (m/s)



Contour: Pressure (Pa)




```
methods (Access = public)
```

```
function this = FuelOilExchangerBlock(tag) ...
function result = RunPhysicalModel(this) ...
function result = LoadPhysicalModel(this) ...
function result = SavePhysicalModel(this)
    result = true;
try
    mphsave(this.Model, fullfile(this.path_file, strcat(this.file
catch exception
    this.Model = [];
    result = false;
    display (sprintf('FuelOilExchangerBlock:SavePhysicalModel -
end
end
function result = InvokePhysicalModel(this)
try
    if isempty(this.Model)
        this.Model = FuelOilExchangerBlock.PhysicalModel;
        this.engine_oil_temperature = str2double(this.PhysicalValue
        this.engine_oil_mass_flow = str2double(this.PhysicalValuePa
        this.engine_oil_pressure = str2double(this.PhysicalValuePa
        this.kerosene_temperature = str2double(this.PhysicalValuePa
        this.kerosene_mass_flow = str2double(this.PhysicalValueParse
        this.kerosene_pressure = str2double(this.PhysicalValueParse
    else
        engine_oil_temperature_string = sprintf('%d[degC]', this.eng
        this.Model.param.set('T_engine_oil', engine_oil_temperature
        engine_oil_mass_flow_string = sprintf('%d[kg/h]', this.engin
        this.Model.param.set('m_dot_engine_oil', engine_oil_mass flo
        engine_oil_pressure_string = sprintf('%d[kPa]', this.engine
        this.Model.param.set('p_engine_oil', engine_oil_pressure str
        kerosene_temperature_string = sprintf('%d[degC]', this.keros
        this.Model.param.set('T_kerosene', kerosene_temperature str
        kerosene_mass_flow_string = sprintf('%d[kg/h]', this.keroser
        this.Model.param.set('m_dot_kerosene', kerosene_mass_flow string);
        kerosene_pressure_string = sprintf('%d[kPa]', this.kerosene_p
        this.Model.param.set('p_kerosene', kerosene_pressure_string);

        this.total_energy_net = mphglobal(this.Model, 'ht.fluid1.ntefluxInt', 'dataset', 'dset1');
        this.engine_oil_fluid_inlet_energy = mphglobal(this.Model, 'ht.temp1.ntefluxInt', 'dataset',
        this.engine_oil_fluid_outlet_energy = mphglobal(this.Model, 'ht.of11.ntefluxInt', 'dataset',
        this.kerosene_fluid_inlet_energy = mphglobal(this.Model, 'ht.temp2.ntefluxInt', 'dataset', 'ds
        this.kerosene_fluid_outlet_energy = mphglobal(this.Model, 'ht.of12.ntefluxInt', 'dataset', 'ds

    end

    this.UpdateDialog;
    result = true;
catch exception
    this.Model = [];
    result = false;
    display (sprintf('FuelOilExchangerBlock:InvokePhysicalModel - %s', exception.message));
end
end
```

Block Parameters: Fuel Oil Exchanger Block

Subsystem (mask)

Object name : Fuel Oil Exchanger Block

Object tag : FOEB

Server

Server Port : 2036

Connection

Disconnection

Physical Model

Path : C:\Workshop\RTJT\Analysis\Comsol\Special Exchanger\Fuel Oil Exchanger

File : fuel_oil_tube_exchanger_p1_c6

Clear

Invoke

Load

Ready

Save

Run

Parameters

Parameter Energy

Total Net Energy [W] : 140.5998

Engine Block Parameters: Fuel Oil Exchanger Block

Subsystem (mask)

Engine

Kerosene

Object name : Fuel Oil Exchanger Block

Object tag : FOEB

Server

Server Port : 2036

Connection

Disconnection

Physical Model

Path : C:\Workshop\RTJT\Analysis\Comsol\Special Exchanger\Fuel Oil Exchanger

File : fuel_oil_tube_exchanger_p1_c6

Clear

Invoke

Load

Ready

Save

Run

Parameters

Parameter Energy

Pressure Outlet [kPa] : 210

Temperature Inlet [deg C] : 145

Mass Flow Inlet [kg/h] : 50

Temperature Inlet [deg C] : 10

Mass Flow Inlet [kg/h] : 310

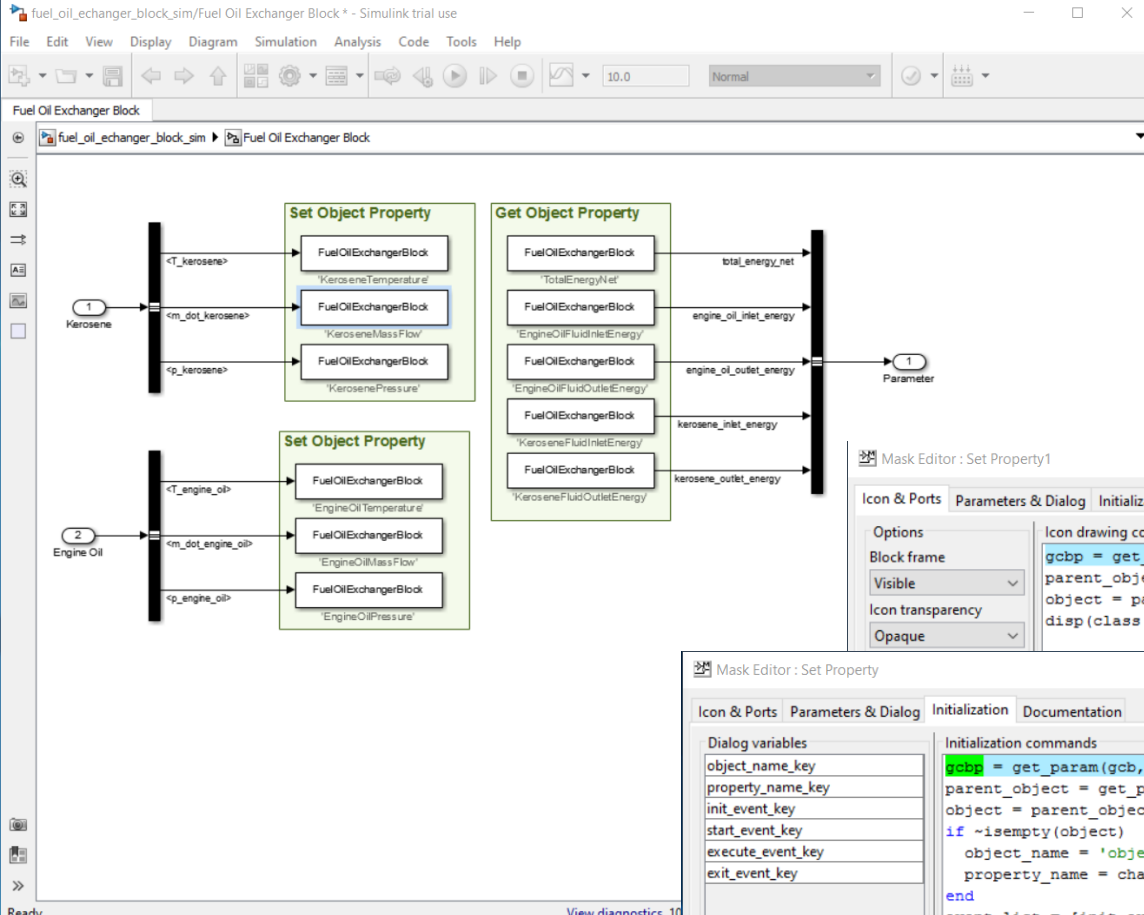
Pressure Outlet [kPa] : 150

OK

Cancel

Help

Apply



Block Parameters: Set Property

S-Function (mask)

Property : 'KeroseneTemperature'

Operation Event :

- Init
- Start
- Execute
- Exit

OK Cancel Help Apply

Mask Editor : Set Property1

Icon & Ports Parameters & Dialog Initialization Documentation

Options

Block frame Visible Icon transparency Opaque

Icon drawing commands

```
gcbp = get_param(gcb, 'Parent');
parent_object = get_param(gcbp, 'UserData');
object = parent_object.Block;
disp(class(object));
```

Mask Editor : Set Property

Icon & Ports Parameters & Dialog Initialization Documentation

Dialog variables

- object_name_key
- property_name_key
- init_event_key
- start_event_key
- execute_event_key
- exit_event_key

Initialization commands

```
gcbp = get_param(gcb, 'Parent');
parent_object = get_param(gcbp, 'UserData');
object = parent_object.Block;
if ~isempty(object)
    object_name = 'object';
    property_name = char(property_name_key);
end
event_list = [init_event_key == 1, ...
              start_event_key == 1, ...
              execute_event_key == 1, ...
              exit_event_key == 1];
```

Allow library block to modify its contents

Unmask Preview OK Cancel Help Apply

```
61 free(_property_name);
62 mxFree(_instruction);
63
64 return;
65 }
66
67 sprintf(_instruction, "%s.%s", _object_n
68 _rhs = mxCreateString(_instruction);
69
70 status = mexCallMATLAB(1, &_lhs, 1, &_rh
71
72 memcpy(_y, mxGetPr(_lhs[0]), sizeof(doub
73
74 free(_object_name);
75 free(_property_name);
76 mxFree(_instruction);
77 }
```

Command Window



Děkuji za pozornost