ROSMOSE: A Web-based Tool for Pinch Analysis and Utility Integration

Advanced Energetics Team

2024

IPESE
Industrial Process and Energy Systems Engineering

École polytechnique fédérale de Lausanne

ROSMOSE
General information

- ROSMOSE:
  - Newly developed web-based tool using Pinch Analysis methods
  - User friendly interface to simplify utilization of OSMOSE framework
- OSMOSE:
  - Used for over two decades in research projects and teaching activities
  - Metaheuristic and deterministic approaches for combined heat and mass integration

Reference: Open-source scientific publishing system https://quarto.org/ , https://rmarkdown.rstudio.com/

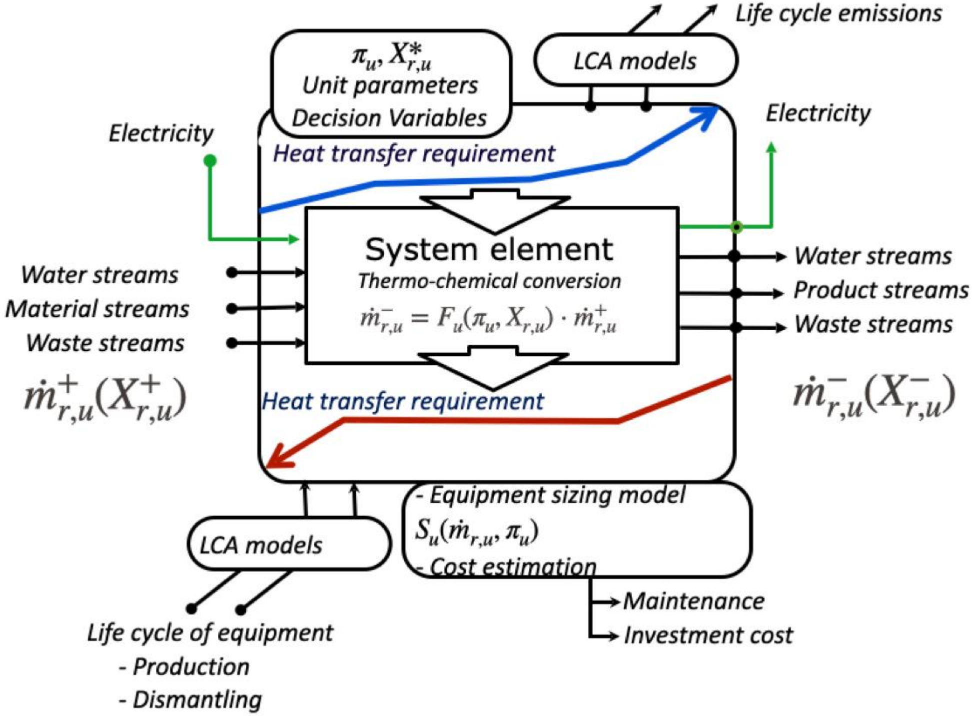
ROSMOSE
General information

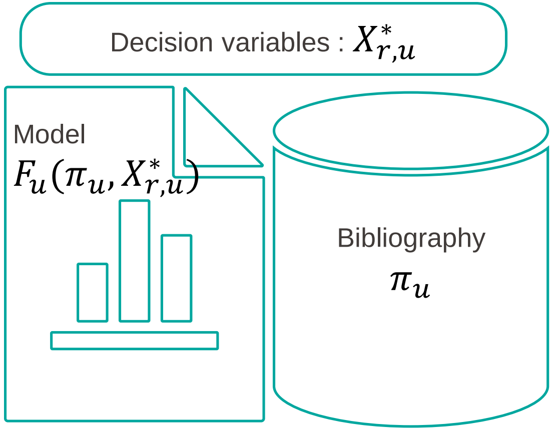
- Accessible through dynamic development environments
- Flexible in integrating various software and tools
- Many possible output formats for standard and interactive reports

Make Science Reproduceable!

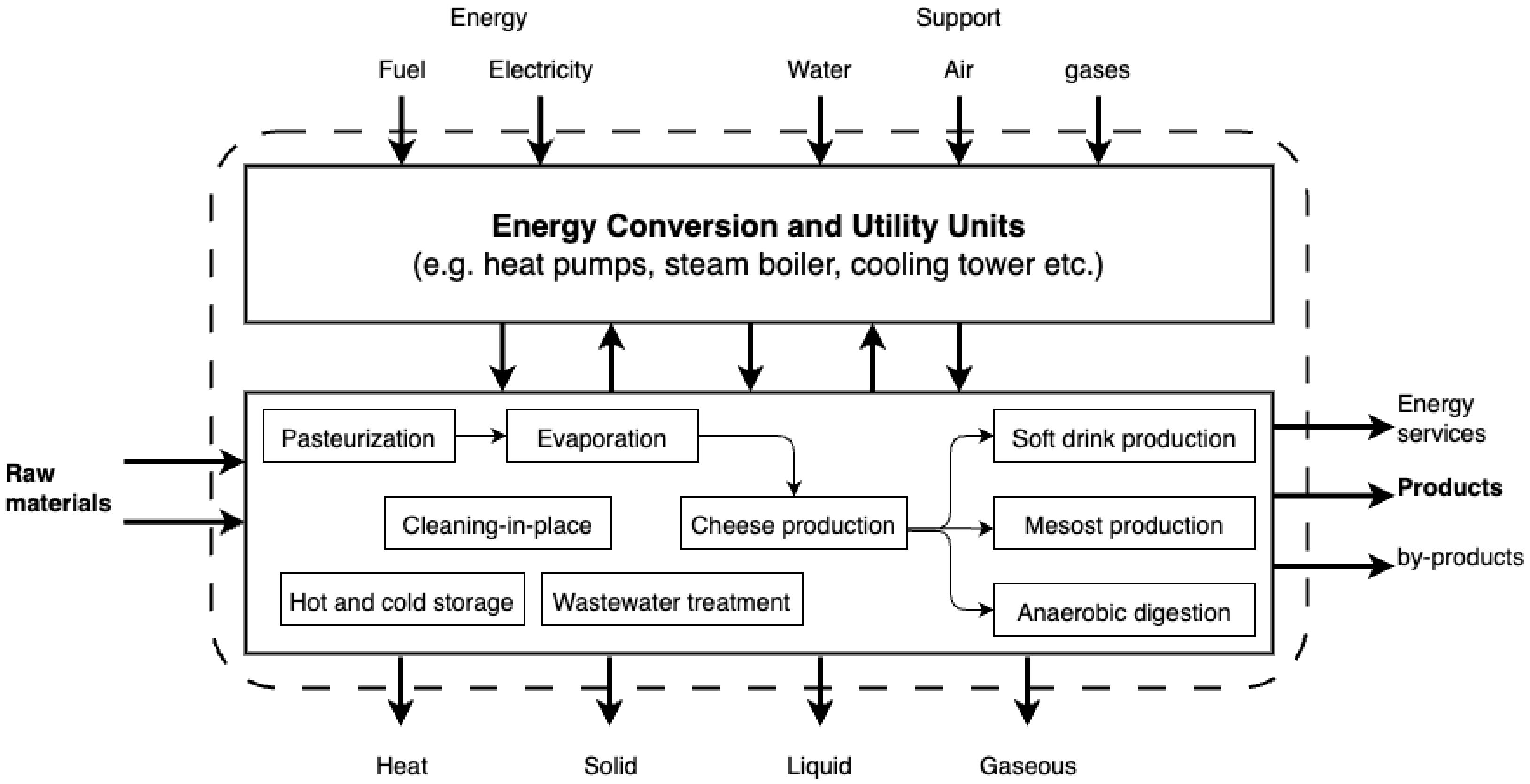
Reference: R Mardown from R Studio, https://rmarkdown.rstudio.com/

ROSMOSE
Process integration/technology blueprints



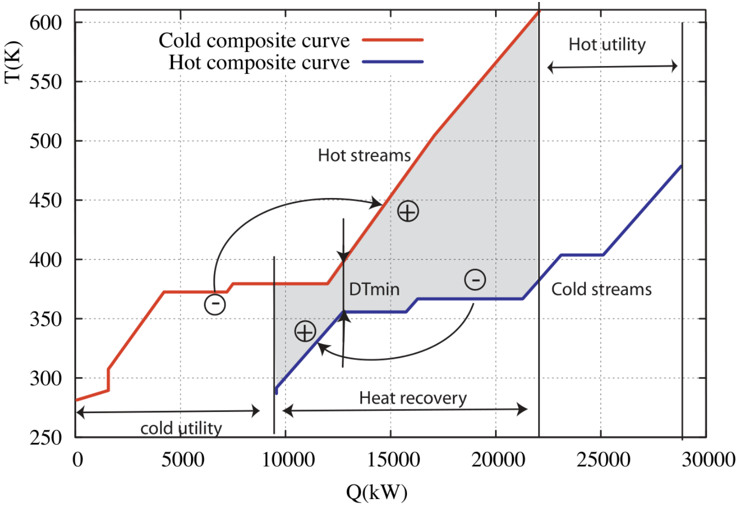


ROSMOSE
Process integration/technology blueprints



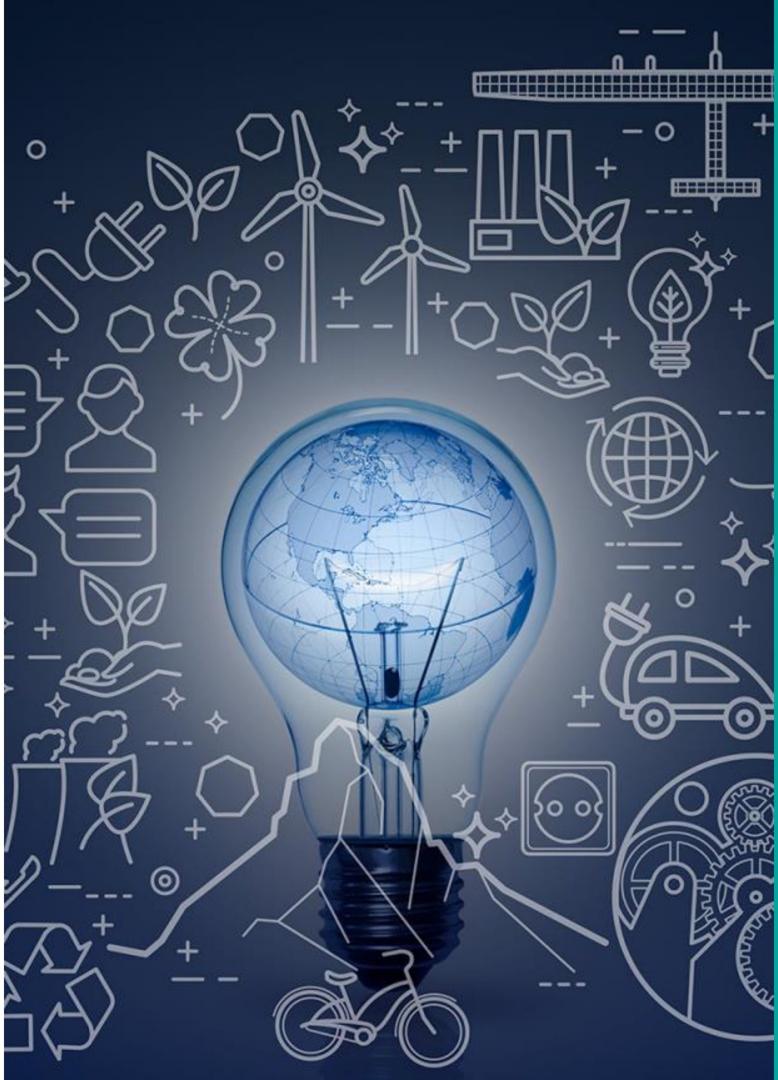
ROSMOSE
General information

- Allows users to:
  - Define systems and energy technologies
  - Call external software (e. g. Aspen Plus)
  - Perform pinch analysis calculations
  - Solve optimisation problems
  - Integrate utilities
  - Compare operational scenarios
  - Produce dynamic reports or presentations



Hot and cold composite curves of a process

Reference: Maréchal, François. (2010). Advanced Energetics Course Notes



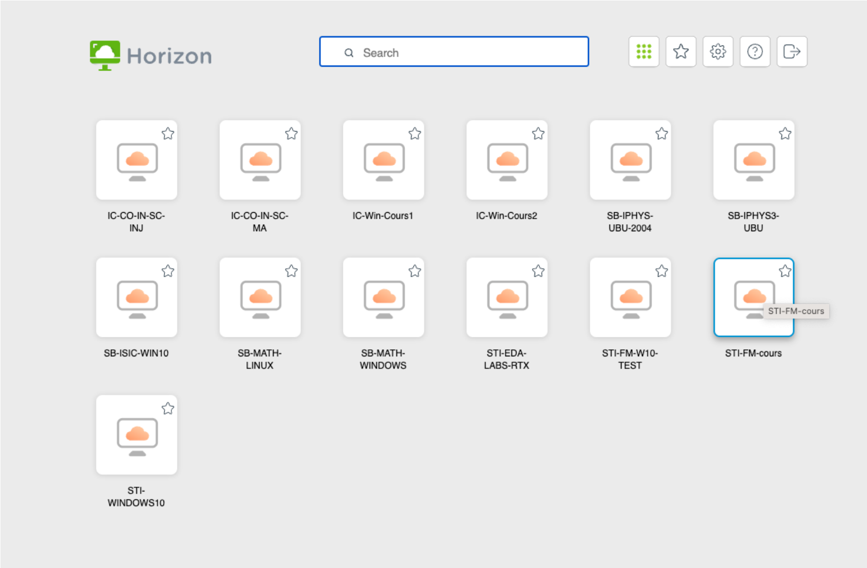
ROSMOSE Tutorial

OUTLINE

Overview of the Quarto book and the environment

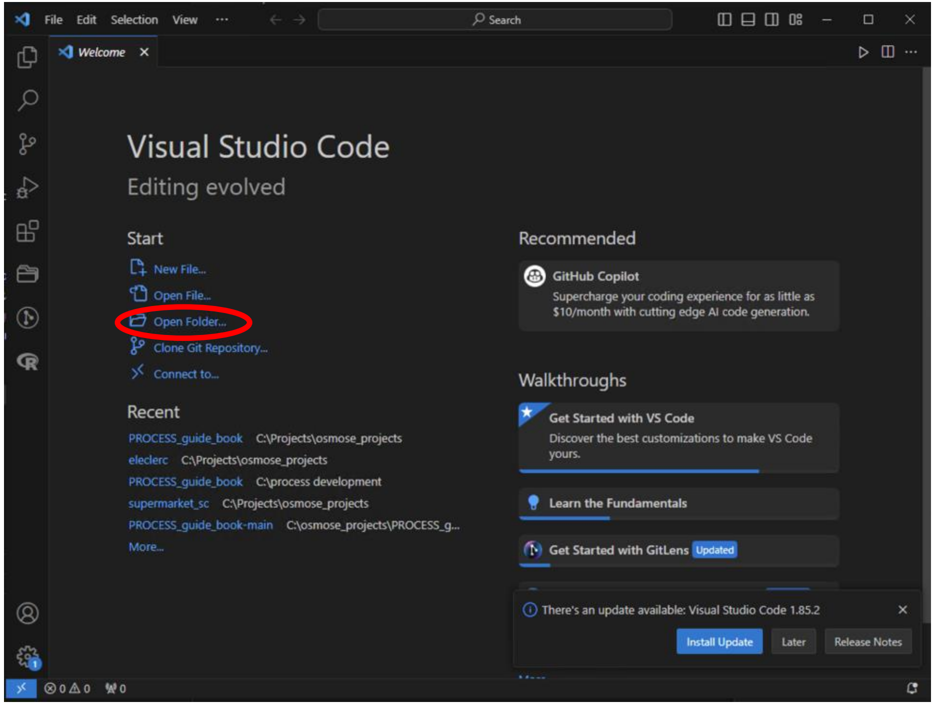
VM Access & File Download

- Access your VM through https://vdi.epfl.ch --> STI-FM-course
- Download the project files from the git repository mentioned in Moodle
- Move the files to the VM desktop and extract



Open the Project Files

- Open Visual Studio Code
- Open the file using "open folder"



- Then "select folder"

The quarto book is now open in VSCode !

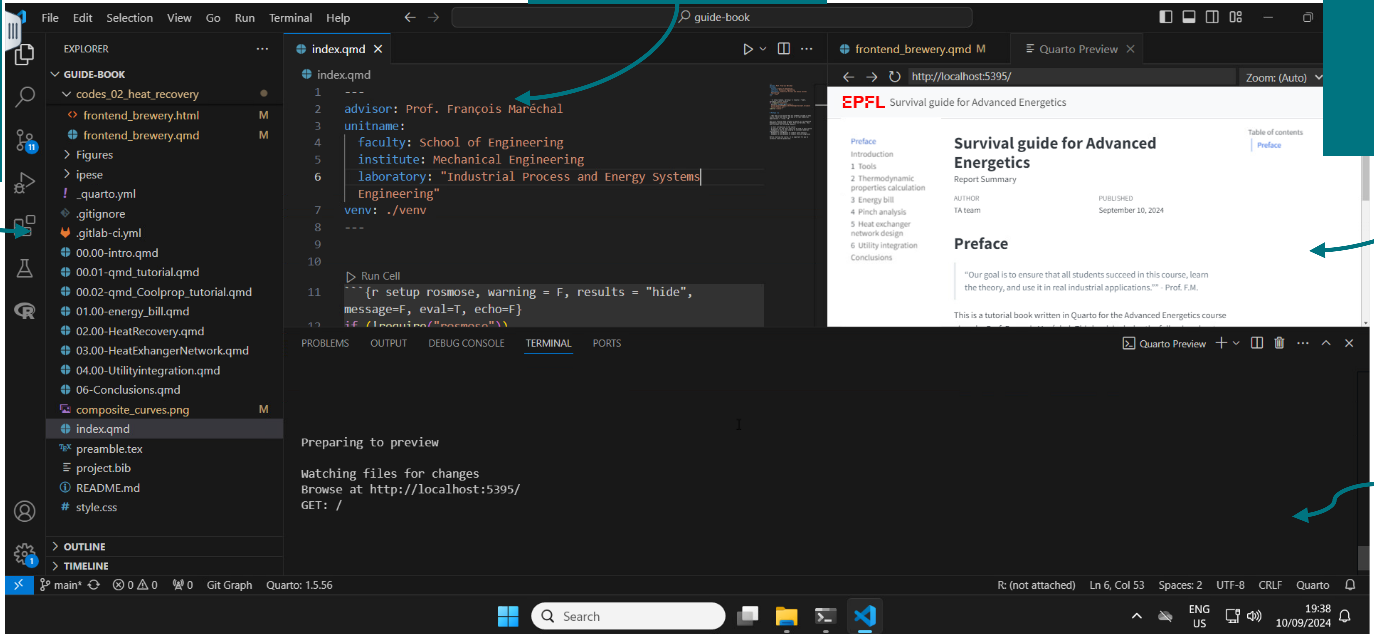
Visual Studio Code environment

This is where are the files of your models and results

This is the environnement, where you can code

A preview of your report

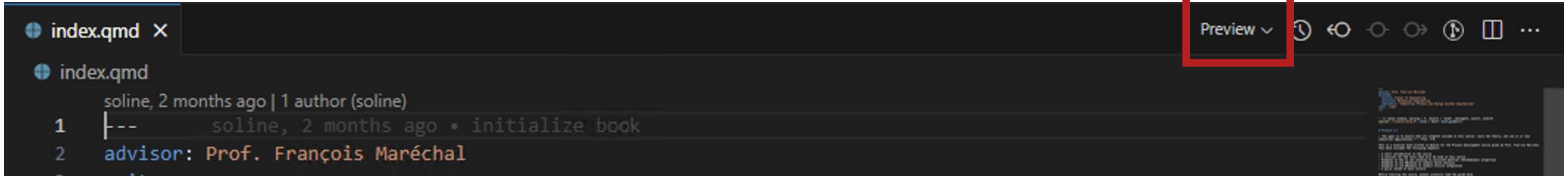
This is the terminal, where you can execute your code



Building the quarto book

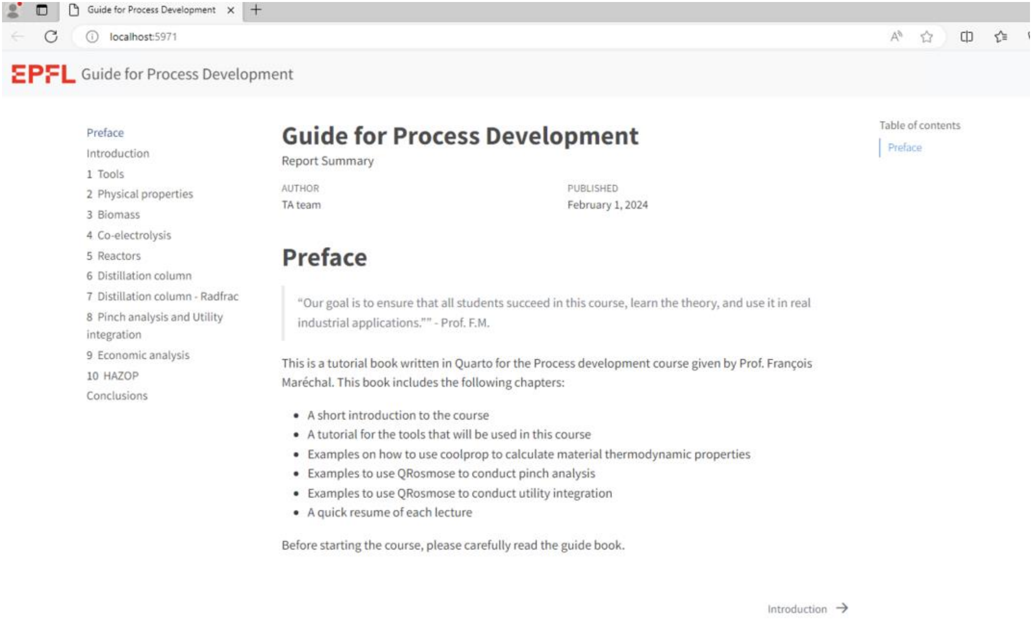
- Open the index.qmd file

Click on “preview”

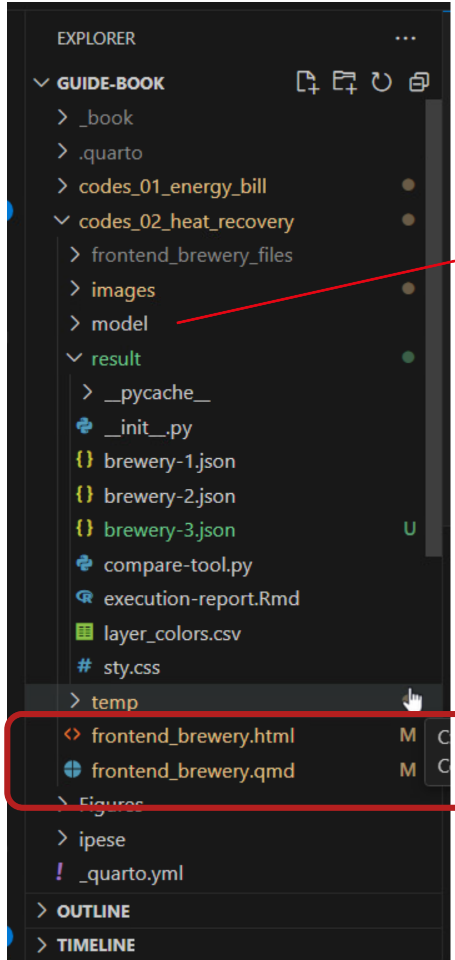


Or write *quarto preview* in the terminal

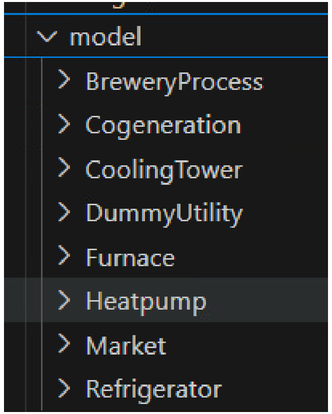
It creates an html book



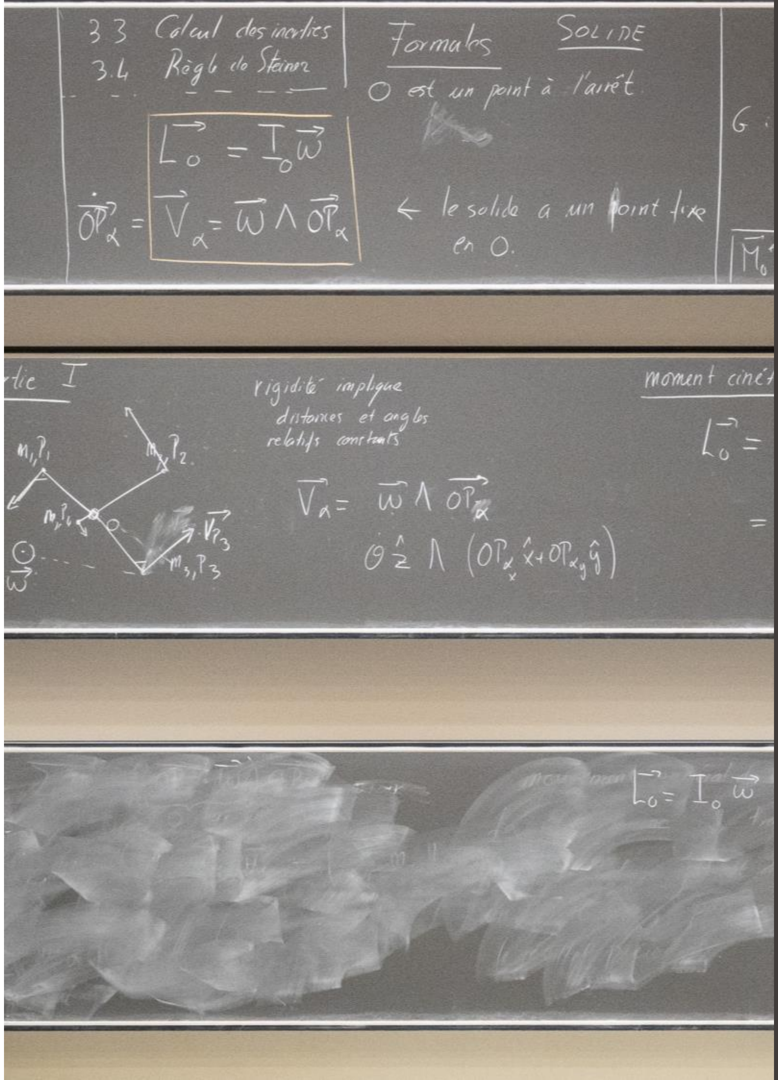
Overview of a quarto book - composition



The folder *Model* contains ROsmose files



The Frontend: *MER* (minimum energy requirement) and *Total Cost*, they allow you to run the model



OUTLINE

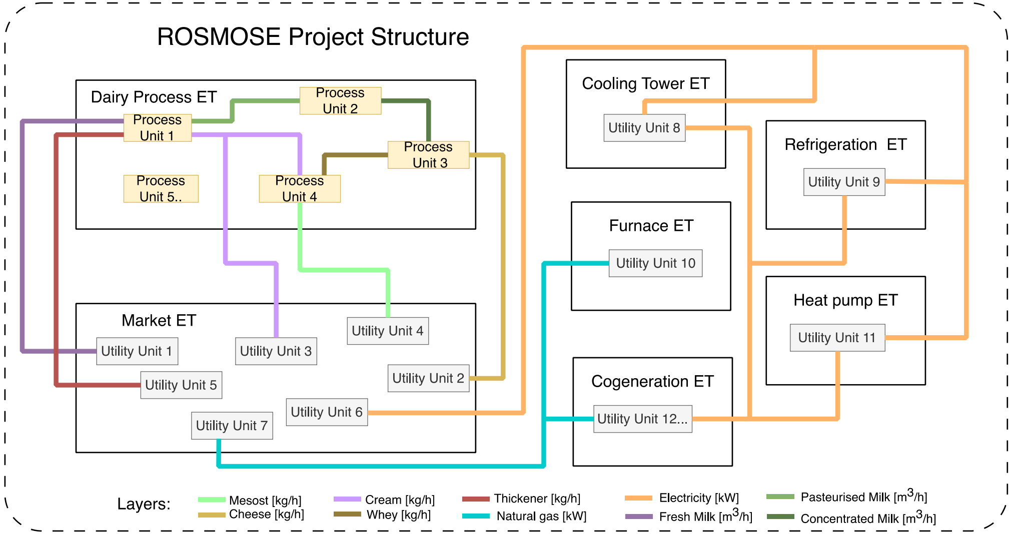
Concept

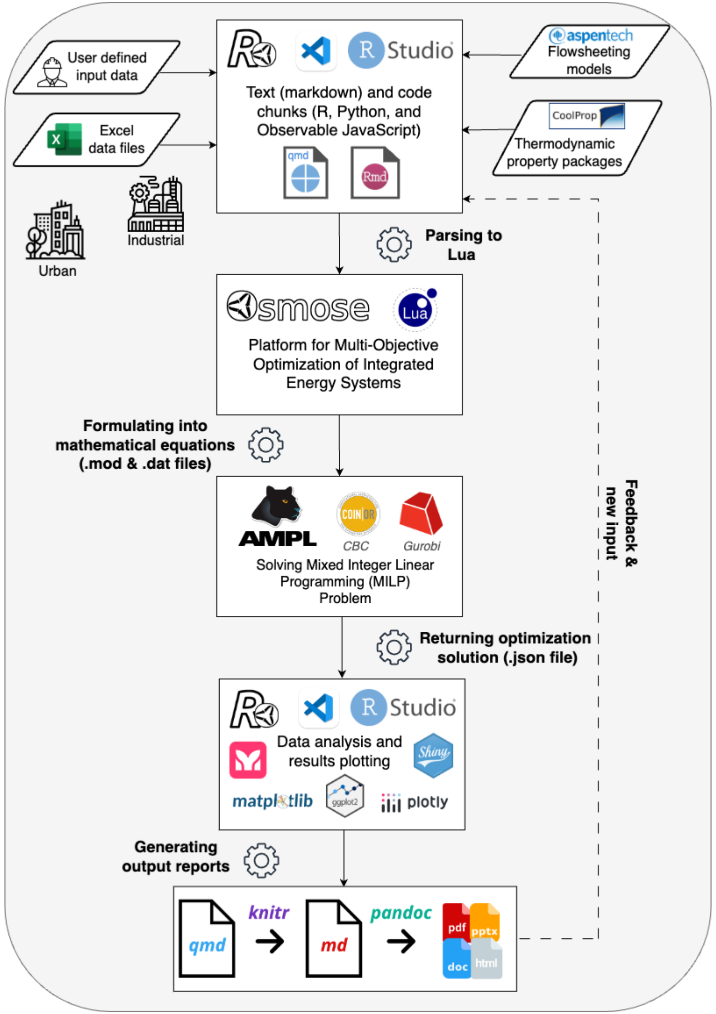
Functionalities

Case study

Conclusions

ROSMOSE Structure





ROSMOSE Tutorial

ROSMOSE Structure

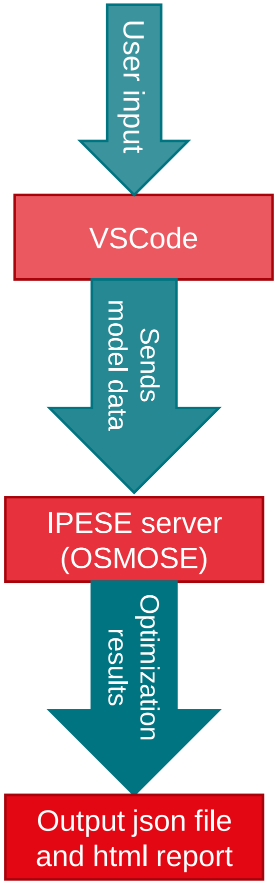
- ROSMOSE variable is defined as a *tag*, i.e. an object containing:
  - a name,
  - a value,
  - a physical unit, and
  - a description

- Tags are used to specify: inputs, outputs, or parameters of a model
- Building blocks of an ROSMOSE project are called *energy technology (ET) models*
- Layers in ROSMOSE can be thought of as the ‘pipes’ through which similar streams flow and are described by a name and a physical unit

Using Rosmose

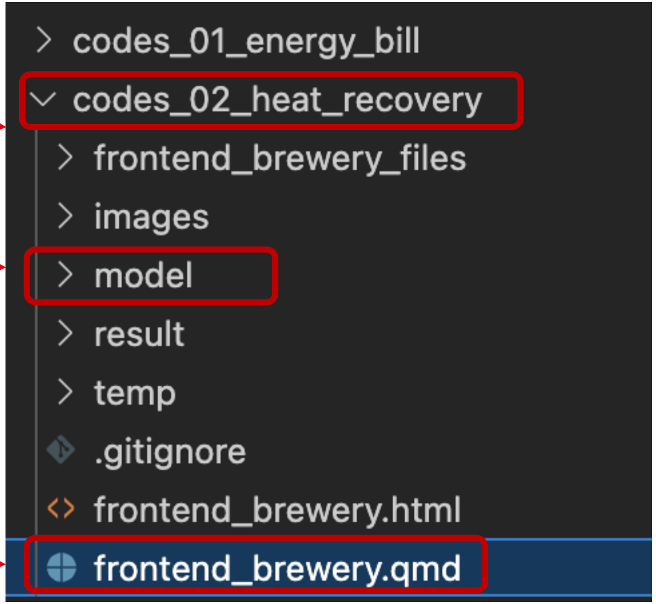
- You already have a running example in the folder "Rosmose"

1. User defines the model to optimize
2. User asks the IPESE server to solve the optimization problem
3. IPESE tools process the models and solve the problem
4. If the calculation succeeds, the results are sent in a .JSON file
5. An automatic .html report is also generated summarizing model results and sample plots



Using Rosmose

- Heat recovery codes folder where you have everything related to perform and report your pinch analysis
- Models: definition of energy technologies, units, processes, resource layers, …
- The frontend is the main script that dictates what to include in the optimization and run it.



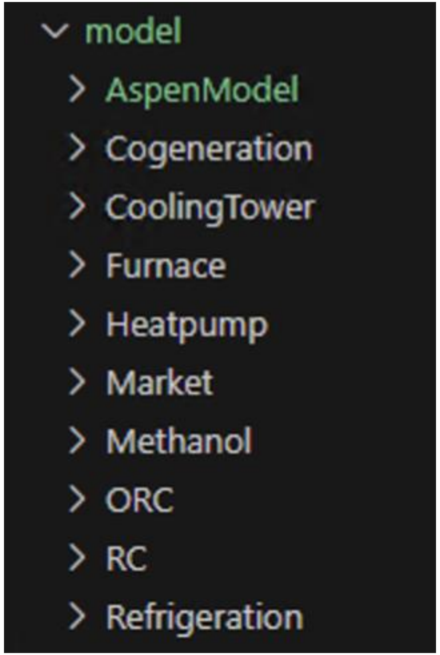
Description of the ETs

Using Rosmose – Content of model folder

- In the model folder, you define the model for your Energy Technologies (ET):

  - The ETs *Cogeneration, Cooling Tower*, *Furnace, Heatpump, Methanol, ORC (Organic Rankine Cycle), RC (Rankine Cycle) and Refrigeration*
  - The *market,* to close the mass and energy balance with the resources consumed

- For the first step, you will only model your processes to determine the minimum energy requirements and estimated savings



Definition of the ETs

Name the model

Defining your layers

Defining process units

```
# Biomass ET {-}

```{rosmose Methanol}
! OSMOSE ET Methanol
```
This ET will use the following Layers

```{rosmose Methanol_layers}
: OSMOSE LAYERS Methanol

|Layer     |Display name   |shortname  |Unit       |Color  |
|:---------|:--------------|:----------|:----------|:------|
|ELEC      |Electricity    |elec       |kW         |yellow |

```

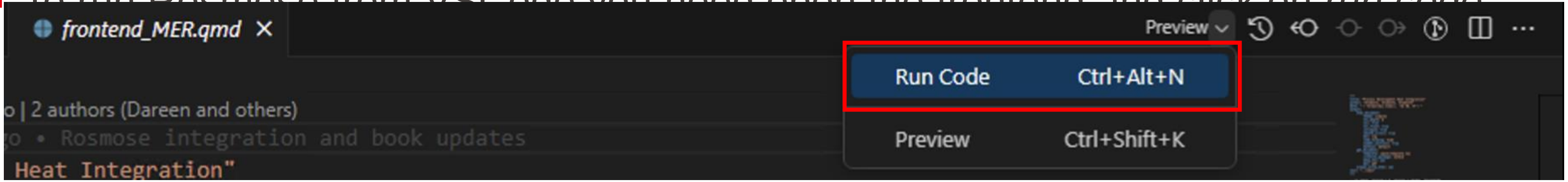

The methanol ET contains the following units

```{rosmose Methanol_units}
: OSMOSE UNIT Methanol

|unit name  |type   |
|:----------|:------|
|Methanol   |Process|
```
```

The inputs

You can also define tags and use them in your model definitions. Example here for Heatpump1. A tag has a name, a value, a unit and a definition

````
```{rosmose}
Evap_Tin = 54.5 [C] # Evaporator temperature inlet
Evap_Tout = 54.5 [C] # Evaporator temperature outlet
Cond_Tin = 76.5 [C] # Condenser temperature inlet
Cond_Tout = 76.5 [C] # Condenser temperature outlet
Cond_Qmax = 6000 [kW] # Condenser reference heat flow rate
exeff = 0.5 [-] # Second law efficiency
dtmin_2ph = 2 [C] # phase-change delta t minimum
n = 40.0 [yr] #lifetime of a heat pump
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

Note: Heat pumping will be profitable only if it allows to transfer heat from below to above the pinch, i.e.

```{rosmose}
COPcarnot = (%Cond_Tin% + 273) / (%Cond_Tin% - %Evap_Tin%) [-] # Carnot COP
COP = %exeff% * %COPcarnot% [-] # Actual COP
W_heatpump = %Cond_Qmax% / %COP% [kW] # Heat pump power consumption
Evap_Qmax = %Cond_Qmax% * (%COP% - 1) / %COP% [kW] # Evaporator heat flow rate
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
Cinv2_HP = 300*%Cond_Qmax%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y] #300 Euro/kWth at the condenser
```
````

Parameters of ETs

- We have to define some parameters for our ET

~~~
```{rosmose Methanol_params}
: OSMOSE UNIT_PARAM Methanol

|cost1|cost2|cinv1|cinv2|imp1|imp2|fmin|fmax|
|:----|:----|:----|:----|:---|:---|:---|:---|
|0    |0    |0    |0    |0   |0   |1   |1   |
```
~~~

- And the hot and cold streams in our processes

~~~
**Heat Streams**

```{rosmose Methanol_hs}
: OSMOSE HEAT_STREAMS Methanol

|name     |Tin         |Tout         |Hin  |Hout        |DT min/2|alpha|
|:--------|:-----------|:------------|:----|:-----------|:-------|:----|
|heater_1 |%T_AIR_1%   |%T_AIR_2%    |0    |%heater_1%  |2.5     |1    |
|heater_3 |%T_FUEL_1%  |%T_FUEL_2%   |0    |%heater_3%  |2.5     |1    |
|heater_4 |%T_FUEL_2%  |%T_FUEL_3%   |0    |%heater_4%  |2.5     |1    |
|heater_5 |%T_FUEL_3%  |%T_FUEL_4%   |0    |%heater_5%  |2.5     |1    |
|cooler_1 |%T_AIR_OUT% |%T_AIR_4%    |0    |%cooler_1%  |2.5     |1    |
|cooler_2 |%T_FUEL_OUT%|%T_LTFUEL%   |0    |%cooler_2%  |2.5     |1    |
|MSC_cooler_1 |%MSC_cooler_1_Tin% |%MSC_cooler_1_Tout% |0    |%MSC_cooler_1_Duty%  |2.5      |1    |
|MSC_cooler_2 |%MSC_cooler_2_Tin% |%MSC_cooler_2_Tout% |0    |%MSC_cooler_2_Duty%  |2.5      |1    |
|HX_1     |%T_S1%       |%T_S2%       |0    |%HX1_Duty%  |2.5     |1    |
|HX_2     |%T_S4%       |%T_S5%       |0    |%HX2_Duty%  |2.5     |1    |
|HX_3     |%T_S7%       |%T_S8%       |0    |%HX3_Duty%  |2.5     |1    |
|R_1      |%T_S3%       |%T_S4%       |0    |%R1%        |2.5     |1    |
|reb      |%reb_ti%     |%reb_to%     |0    |%reb_Q%     |2.5     |1    |
|cond     |%cond_ti%    |%cond_to%    |0    |%cond_Q%    |2.5     |1    |
~~~

- If your unit is consuming a resource, you can define it as follow (example for the power consumption)

~~~
```{rosmose}
Power_consump = %MSC_power_tot%-%Electrolyzer_size%
```

```{rosmose Methanol_rs}
: OSMOSE RESOURCE_STREAMS Methanol

|layer     |direction|value              |
|:---------|:--------|:------------------|
|ELEC      |in       |%Power_consump%    |
```
~~~

Heat streams

````
**Heat Streams**

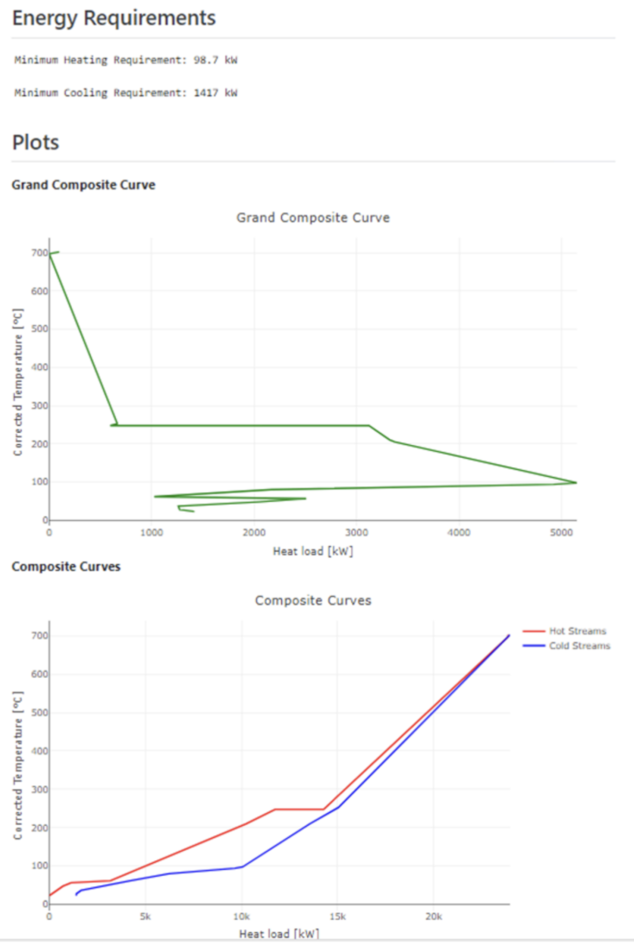
```{rosmose Methanol_hs}
: OSMOSE HEAT_STREAMS Methanol

|name     |Tin          |Tout          |Hin  |Hout         |DT min/2|alpha|
|:--------|:------------|:-------------|:----|:------------|:-------|:----|
|heater_1 |%T_AIR_1%    |%T_AIR_2%     |0    |%heater_1%   |2.5     |1    |
|heater_3 |%T_FUEL_1%   |%T_FUEL_2%    |0    |%heater_3%   |2.5     |1    |
|heater_4 |%T_FUEL_2%   |%T_FUEL_3%    |0    |%heater_4%   |2.5     |1    |
|heater_5 |%T_FUEL_3%   |%T_FUEL_4%    |0    |%heater_5%   |2.5     |1    |
|cooler_1 |%T_AIR_OUT%  |%T_AIR_4%     |0    |%cooler_1%   |2.5     |1    |
|cooler_2 |%T_FUEL_OUT%|%T_LTFUEL%     |0    |%cooler_2%   |2.5     |1    |
|MSC_cooler_1 |%MSC_cooler_1_Tin% |%MSC_cooler_1_Tout% |0    |%MSC_cooler_1_Duty%  |2.5     |1    |
|MSC_cooler_2 |%MSC_cooler_2_Tin% |%MSC_cooler_2_Tout% |0    |%MSC_cooler_2_Duty%  |2.5     |1    |
|HX_1     |%T_S1%       |%T_S2%        |0    |%HX1_Duty%   |2.5     |1    |
|HX_2     |%T_S4%       |%T_S5%        |0    |%HX2_Duty%   |2.5     |1    |
````

- **Definition of input heat streams**
  - **Tin:** stream inlet temperature [℃]
  - **Tout:** stream outlet temperature [℃]
  - **Hin:** stream inlet enthalpy (0 reference point) [kW]
  - **Hout:** stream outlet enthalpy (Q for hot and cold streams) [kW]
  - **DTmin/2:** Dtmin is the minimum temperature difference allowed for the heat exchange
  - **alpha:** stream heat transfer coefficient [kW/m²/K]
  - Look at the lecture notes for a guidelines on Dtmin/2 and alpha for different types of streams!
  - Pay attention to only put letters and numbers in the name of the streams. **No special characters (_ - ; …)**
````

# ROSMOSE
# MILP solver

- OSMOSE Lua backend framework handles solving the mixed integer linear programming (MILP) problem

$$min\ Cost_{tot} = \sum_{u \in Units} \left[ \sum_{t \in Time} \left( Cop1_u \cdot Y_{u,t} + Cop2_u \cdot M_{u,t} \right) \cdot \Delta t + Cinv1_u \cdot Y_u + Cinv2_u \cdot M_u \right]$$

- Subject to multiple constraints:

$$Y_{u,t} \cdot Fmin_u \leq M_{u,t} \leq Y_{u,t} \cdot Fmax_u \qquad ; \forall u \in Units, \qquad \forall t \in Time$$

$$\sum Input\ streams = \sum Output\ streams \qquad ; \forall\ Layer, \qquad \forall\ Time$$

$$\sum_{w=1}^{N_w} M_w \cdot q_{w,r} + \sum_{i=1}^{N} Q_{i,r} + R_{r+1} - R_r = 0 \qquad ; \forall\ r = 1 \ldots\ldots N$$

# The frontend and the results

# The frontend

- The frontend is the main script that dictates what to include in the optimization and run it.
- It the file that will be compiled by quarto and will launch the optimization
  - It is your optimization scenario
  - You can define multiple frontends corresponding to multiple scenarios (here minimum energy requirement and cost optimization)
  - Generates two outputs: an optimization report in html and a json results file
- To run Rosmose from VSCode you need open the frontend and click on *run code*




- This command will automatically run all lines in your frontend.Rmd file:
  - Importing Rosmose
  - Importing the models of your ETs
  - Plotting your optimization results and generating the output html report

# ROsmose Solve Options

- Osmose-solve options
  - MER: Minimum energy requirements (only process targets)
  - TotalCost: optimal integration of utility units
  - With an operating time of 8760 h/year

````
```{rosmose solve-osmose}
! OSMOSE SOLVE-LOCAL methanol MER [Methanol]


|name    |value|
|:------|:----|
|op_time|8760 |
```
````

# Minimum energy requirement – MER frontend

- After running the frontend, you can see the results in your browser
- It has plotted the grand composite curve, the composite curves and the MER calculation



# Frontend MER in ROSMOSE


## Process Development Heat Integration

AUTHOR
Student1, Student2, Student3

PUBLISHED
February 1, 2024

```
source("../rosmose-setup.R", local = knitr::knit_global())
```

## Introduction

This report presents the results of your heat integration simulations. Each run will generate an output json file stored in the “Rosmose/result” folder and a “frontend.html” file stored in “Rosmose” folder.

The json files will be numbered as 1, 2, 3,,,,,, for each run. However, the “frontend.html” file is overwritten each time hence to keep previous versions you must rename the file before running your simulation again.

once your models and units are ready you can run the rsomose simulation by pressing the “Preview” button in the top right corner or using the command line “quarto preview” in the terminal at the correct directory location.

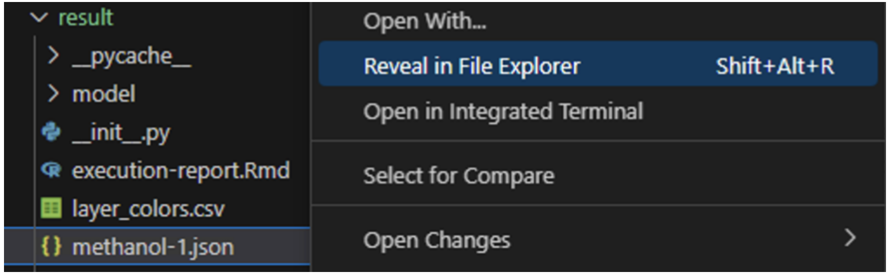
A ready example is given to demonstrate extraction of data from an Aspen flowsheet, changing the process parameters in Aspen through your Rmd file and displaying your integration results.

all Aspen related files are found in this folder “model/AspenModel/.”

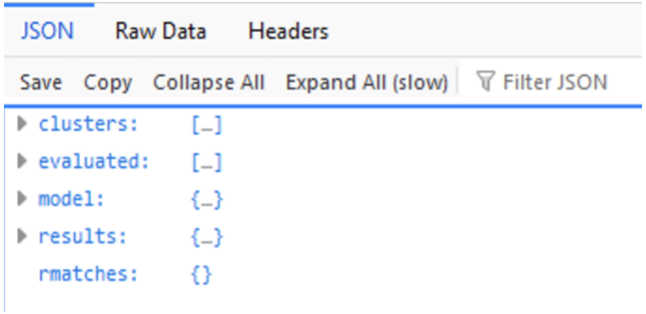
The “Aspen_methanol_process.rmd” file is coupled with the “methanol_process_PD1.bkp” file. The same example Aspen process flowsheet used in all tutorials throughout the course.

# Total cost frontend

- The results are in the folder *Results,* in the *methanol.json* file. Click right and open in file explorer

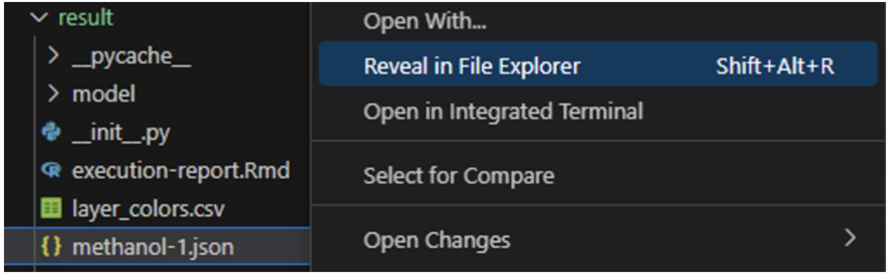



- Open the file with Firefox.

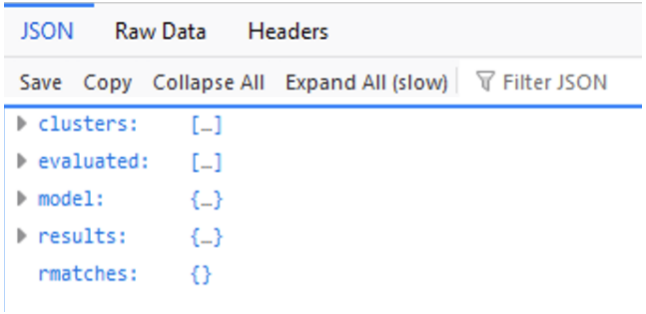


# Total cost frontend

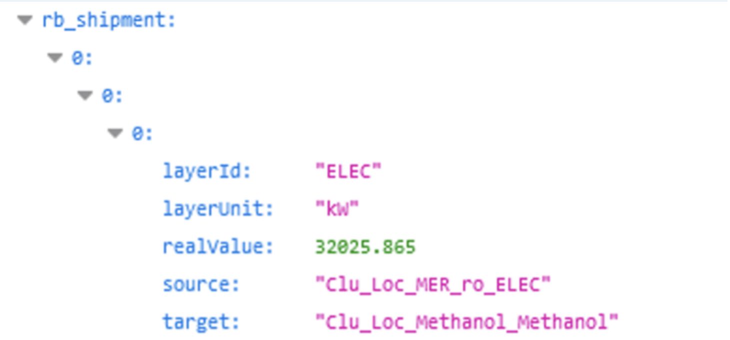
- The results are in the folder *Results*, in the *methanol.json* file. Click right and open in file explorer




- Open the file with Firefox.




- The results are displayed. The main ones you will need are in results > rb_shipment. It is the flows and their value.



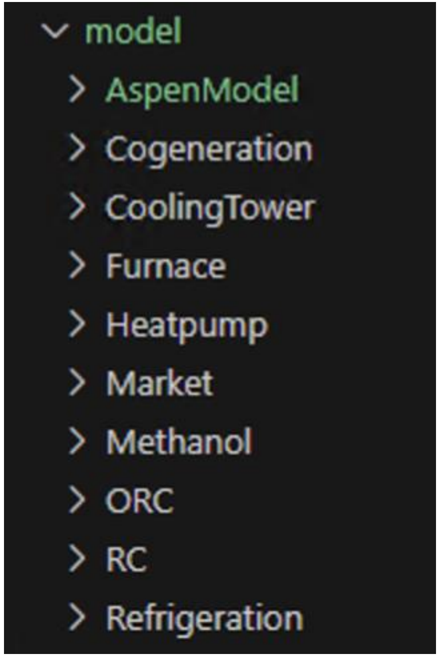

- You can also find the KPIs as the capex, the impact, the opex…


OUTLINE

# The utilities

# Using Rosmose – definition of the utilities

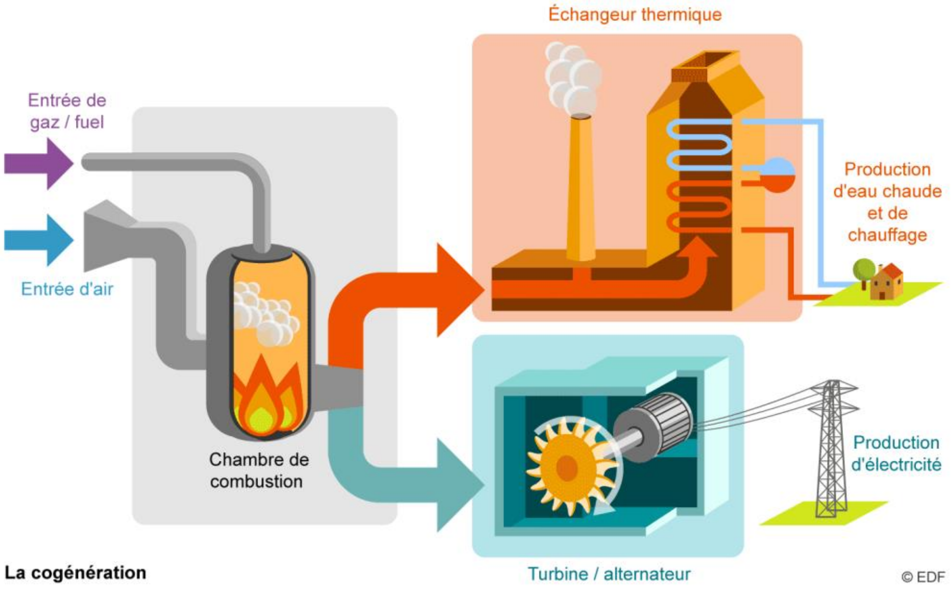
- In the model folder, you define the model for your Energy Technologies (ET). In our case, we have 9 ETs:
  - A sample Aspen Model demonstrating how to import flowsheet data from Aspen into ROSMOSE
  - The ETs *Cogeneration, Cooling Tower*, *Furnace, Heatpump, Methanol, ORC (Organic rankine cycle), RC (Rankine Cycle) and Refrigeration*
  - The *market,* to close the mass and energy balance with the resources consumed




- In this class, we will see what is inside the ETs

# Cogeneration

- Cogeneration consists in producing at the same time electricity and heat (thermic energy)
- Gas and air enters the combustion chamber
- There is a thermic exchanger that allows to produce hot water and heat
- A turbine produces electricity



EPFL

# Cogeneration – Rosmose ET


- Inputs

```
```{rosmose}
! OSMOSE ET cogen
```

```{rosmose}
eta_el = 0.4 #electrical efficiency
eta_th_fg = 0.22  #thermal efficiency - high grade heat if form of flue gases (typically available @ 450 can be cooled down to 150°C)
eta_th_cw = 0.25  #thermal efficiency - low grade waste heat in form of cooling water (@ 90 - 50°C)
Cogen_natGas_LOAD = 6000 [kW] # Reference cogeneration unit load
fg_Tin = 450 [C] #high-grade waste heat (Flue_gas) available temperature
fg_Tout = 150 [C] #high-grade waste heat (Flue_gas) exit temperature
cw_Tin = 90 [C] #low-grade waste heat (cooling_water) available temperature
cw_Tout = 40 [C] #low-grade waste heat (cooling_water) exit temperature
n = 40.0 [yr] #lifetime
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

```{rosmose}
Cogen_elec = %eta_el%*%Cogen_natGas_LOAD% #Power generation assuming ~40% efficiency
Q_cogen_fg = %eta_th_fg% * %Cogen_natGas_LOAD%  #high grade heat generated from flue gases assuming 22%
Q_cogen_cw = %eta_th_cw% * %Cogen_natGas_LOAD%  #high grade heat generated from flue gases assuming 25%
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
Cinv2_cogen = 1200*%Cogen_elec%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y] #1200 Euro/kW natural gas load of the cogeneration unit
```
```

# Cogeneration – Rosmose ET




- Layers and parameters

```
**Layers of the Cogeneration ET**

```{rosmose}
: OSMOSE LAYERS cogen

|Layer      |Display name   |shortname  |Unit       |Color  |
|:----------|:--------------|:----------|:----------|:------|
|NATGAS     |Gas            |ng         |kW         |green  |
|ELEC       |Electricity    |elec       |kW         |yellow |

```

**Cogeneration unit of the Cogeneration ET**

```{rosmose}
: OSMOSE UNIT cogen

|unit name  |type     |
|:----------|:--------|
|cogen      |Utility  |
```

**Parameters of the Cogeneration unit**

```{rosmose}
: OSMOSE UNIT_PARAM cogen

|cost1  |cost2  |cinv1  |cinv2          |imp1   |imp2   |fmin   |fmax   |
|:------|:------|:------|:--------------|:------|:------|:------|:------|
|0      |0      |0      |%Cinv2_cogen%  |0      |0      |0      |5      |
```
```

# Cogeneration – Rosmose ET


- Streams

```
**Cogeneration Streams**

*Resource Streams*
Defining the resource streams, in this case natural gas to, and electricity from, the cogeneration unit

```{rosmose}
: OSMOSE RESOURCE_STREAMS cogen

|layer      |direction|value                |
|:----------|:--------|:--------------------|
|ELEC       | out     |%Cogen_elec%         |
|NATGAS     | in      |%Cogen_natGas_LOAD%  |
```

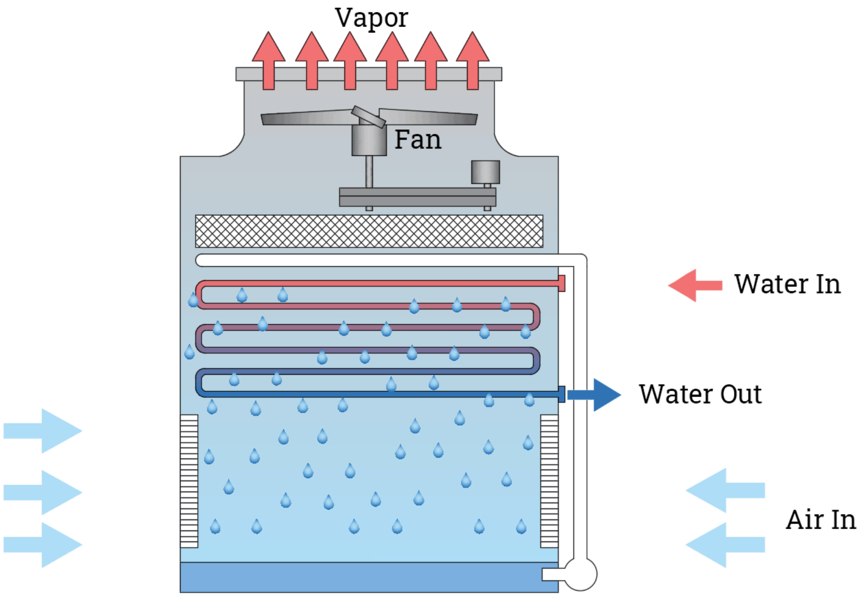
*Heat Streams*

```{rosmose}
: OSMOSE HEAT_STREAMS cogen

|name           |Tin        |Tout       |Hin           |Hout       |DT min/2    |alpha|
|:--------------|:----------|:----------|:-------------|:----------|:-----------|:----|
|fg             |%fg_Tin%   |%fg_Tout%  |%Q_cogen_fg%  |0          |2.5         |1    |
|cw             |%cw_Tin%   |%cw_Tout%  |%Q_cogen_cw%  |0          |2.5         |1    |
```
```

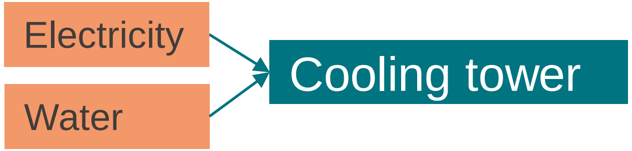
# Cooling tower

- Cooling towers are specialized heat exchangers
- They remove heat from water by latent heat loss from evaporation while coming into contact with an airstream.
- Water is also cooled by sensible heat transfer due to the temperature difference between air and water.
- Cooling towers are basically heat exchangers, generating cooling by bringing water and air into contact.




https://www.iqsdirectory.com/articles/cooling-tower.html

# Cogeneration – Rosmose ET






- Inputs

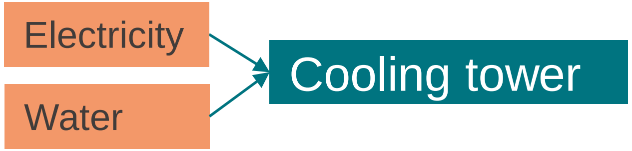
```
# Cooling Tower {-}

```{rosmose coolingtower}
! OSMOSE ET coolingtower
```

```{rosmose}
Cool_Tin = 15 [C] #Cooling tower inlet temperature
Cool_Tout = 30 [C] #Cooling tower outlet temperature
Cool_Qmax = 1000 [kW] #Cooling tower reference heat load
Cool_Elec = 0.021 [kW/kW] #Cooling Tower electricity input kWel/kWth
dtmin_liq = 5 [C] #delta Tmin of the cooling water (w/ liquid streams)
deltaH = 62.8 [kJ/kg] #Enthalpy change for cooling water @1 bar between 15 to 30°C
Twetbulb = 12.17 [C]
n = 40.0 [yr] #lifetime of a cooling tower
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

```{rosmose}
E_ref_CT = %Cool_Elec%*%Cool_Qmax% [kW] # Electricity consumption
deltaT_CT = %Cool_Tout%-%Cool_Tin% [C]
approach = %Cool_Tin%-%Twetbulb% [C]
water_flow = %Cool_Qmax%/%deltaH%*3600 [kg/h] #water flow rate
watermu_CT = 0.000851*%water_flow%*(%Cool_Tout%-%Cool_Tin%) [kg/h] #makeup water in the CT system
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
CTCost = 746.49/0.066*((%water_flow%/1000)**0.79)*(%deltaT_CT%**0.57)*(%approach%**-0.9924)*(0.022*%Twetbulb%+0.39)**2.447 [Euro]
Cinv2_CT = %CTCost%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y]
```
```

# Cogeneration – Rosmose ET




- Layers and parameters

```
**Layers of the Cooling Tower ET**

```{rosmose}
: OSMOSE LAYERS coolingtower

| Layer      | Display name | shortname | Unit | Color  |        Soline Florenc
|:----------|:-------------|:----------|:-----|:------|
|ELEC       |Electricity   |elec       |kW    |yellow |
|WATER      |Water         |water      |kg/h  |blue   |
```

**Cooling tower unit of the Cooling Tower ET**

```{rosmose}
: OSMOSE UNIT coolingtower

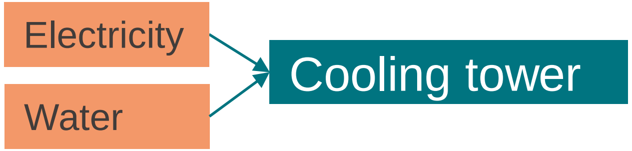
| unit name |type   |
|:----------|:------|
| CoolTower |Utility|
```

**Parameters of the Cooling Tower unit**

```{rosmose CoolTower_params}
: OSMOSE UNIT_PARAM CoolTower

|cost1  |cost2  |cinv1  |cinv2      |imp1   |imp2   |fmin   |fmax   |
|:------|:------|:------|:----------|:------|:------|:------|:------|
|0      |0      |0      |%Cinv2_CT% |0      |0      |0      |100000  |
```
```

# Cogeneration – Rosmose ET




- Streams

```
**Cooling Tower Streams**

Defining the resource streams, in this case electricity to the Cooling Tower

*Resource Streams*


```{rosmose CoolTower_rs}
: OSMOSE RESOURCE_STREAMS CoolTower

|layer      |direction|value               |
|:----------|:--------|:-------------------|
|ELEC       |in       |%E_ref_CT%          |
|WATER      |in       |%watermu_CT%        |
```

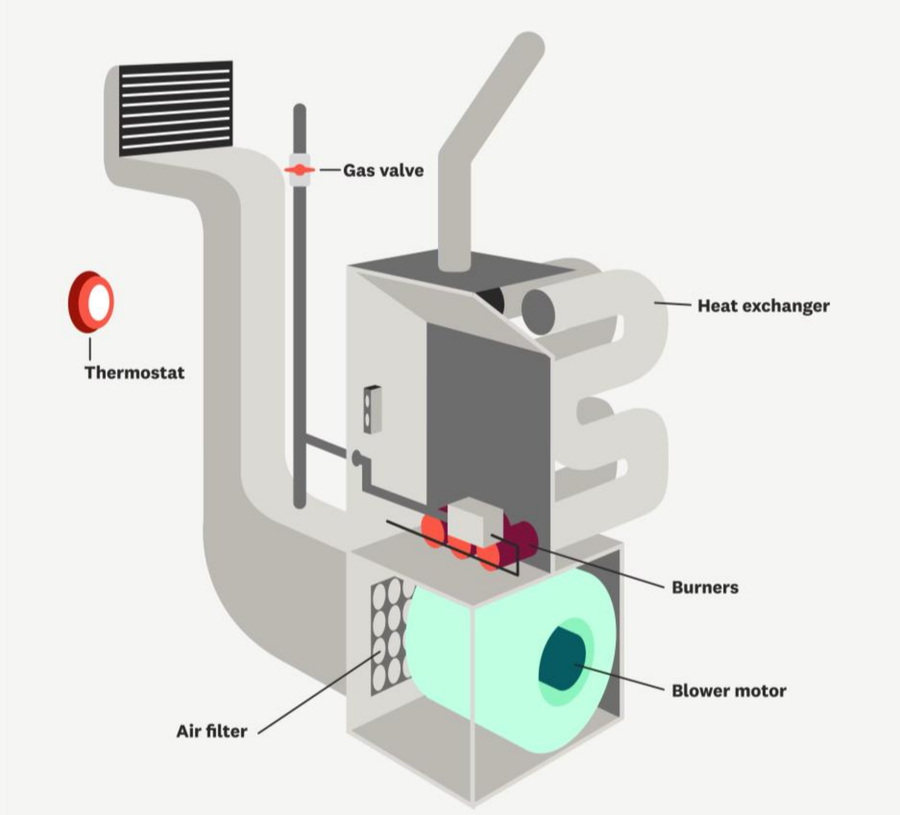
*Heat Streams*

```{rosmose CoolTower_hs}
: OSMOSE HEAT_STREAMS CoolTower

|name          |Tin        |Tout       |Hin |Hout          |DT min/2      |alpha  |
|:-------------|:----------|:----------|:---|:-------------|:-------------|:------|
|cooltowerheat |%Cool_Tin% |%Cool_Tout%| 0  | %Cool_Qmax%  | %dtmin_liq%  |1      |
```
```

# Furnace

- Used to generate heat via a fluid movement (air, steam, hot water)
- Common source for the furnace: natural gas




https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.angi.com%2Farticles%2Fhow-does-it-work-understanding-your-gas-furnace.htm&psig=AOvVaw1VK0y9nBqc24l7Zw2nXYZR&ust=1706888323472000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCKDIptS8ioQDFQAAAAAdAAAAABAY

# Furnace – Rosmose ET

Natural Gas → Furnace → Heat



- Inputs

```
```{rosmose}
Tad = 2025 [C] # Adiabatic flame temperature of the fuel
dtmin_radiation = 2 [C] # radiation delta t mininum
dtmin_convection = 8 [C] # convection delta t mininum
To = 25 [C] # Tamb = To = Tchemicalreference
Trad= 1050 [C] # Radiation temperature threshold, actual temperature is 1050°C but 400 can be used for plot visualization
Tstack = 100 [C] # Stack temperature threshold for no dew point
MWair = 29 [kg/kmol] # Molecular weight of dry air = 79% N2 + 21% O2
MWfuel = 16 [kg/kmol] # Molecular weight of methane
losses = 0.03 [-] # 3% Furnace losses
LHV = 50000 [kJ/kg] # natural gas LHV @ 25°C,1 bar
molratst = 9.52 [kmol/kmol] # Stoich molar air to fuel ratio
a = 1.02 [-] # excess air as in: CH4 + a*2(O2+3.76N2) --> CO2 + 2H2O + 2*3.76N2 + 2(a-1)(O2 + 3.76N2)
cpair = 1.075 [kJ/kg/K] # Air heat capacity @427°C, 1bar (Engineering toolbox)
Tprin = 26 [C] # Preheating temperature
Furnace_natGas_LOAD = 1000 [kW] # Reference furnace load
Spec_heaterCost = 200 [Euro/kW]
n = 40.0 [yr] #lifetime of a furnace
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

```{rosmose}
v = %molratst% * %MWair% / %MWfuel% [kg/kg] # stoich air to fuel mass ratio
cpg = %Furnace_natGas_LOAD% / (%Tad% - %To%) [kW/K] # Flue gases heat capacity @1bar
Tad_corr = %To% + (%Furnace_natGas_LOAD% / (%cpg% + (%cpair% * (%a% -1) * %v% * %Furnace_natGas_LOAD% / %LHV%))) [C] # Corrected adiabatic flame temperature
cpg_corr = %Furnace_natGas_LOAD% / (%Tad_corr% - %To%) [kW/K] # Corrected flue gases heat capacity @1bar
Q_rad_gross = (%Furnace_natGas_LOAD% * (%Tad_corr% - %Trad%) / (%Tad_corr% - %To%)) [kW] # Heat flow rate at the radiation threshold temperature
Q_conv_gross = %Furnace_natGas_LOAD% * (%Trad% - %Tstack%) / (%Tad_corr% - %To%) [kW] # Heat flow rate at the convection threshold temperature
Q_preh = %cpair% * %v% * %Furnace_natGas_LOAD% / %LHV% * (%Tprin% - %To%) [kW] # Air preheating load
Q_stack = %cpg_corr% * (%Tstack% - %To%) [kW] # Stack losses
Q_radpreh = %Q_preh% + %Q_rad_gross% [kW] # Preheating load added to the highest temperature
Q_demand = %Furnace_natGas_LOAD% / (1-%losses%) [kW] # Total energy consumption by the furnace
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
Cinv2_NGFur = %Spec_heaterCost%*%Furnace_natGas_LOAD%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y]      Soline Florence Marie Corre, 6 days ago • ROSMOSE te
```
```

# Furnace – Rosmose ET



Natural Gas → Furnace → Heat

- Layers and parameters

```
**Layers of the Furnace ET**

```{rosmose}
: OSMOSE LAYERS furnace

| Layer         | Display name | shortname | Unit | Color |
|:-------------|:------------|:---------|:----|:-----|
|NATGAS         |Gas           |ng         |kW    |green  |

```

**Furnace unit of the Furnace ET**

```{rosmose}
: OSMOSE UNIT furnace

|unit name  |type   |
|:---------|:-----|
|Furnace    |Utility|
```

**Parameters of the Furnace unit**

```{rosmose Furnace_params}
: OSMOSE UNIT_PARAM Furnace

|cost1  |cost2  |cinv1  |cinv2          |imp1   |imp2   |fmin   |fmax   |
|:-----|:-----|:-----|:-------------|:-----|:-----|:-----|:-----|
|0      |0      |0      |%Cinv2_NGFur% |0      |0      |0      |100    |
```
```

# Furnace – Rosmose ET

Natural Gas → Furnace → Heat

- Streams

```
**Furnace Streams**

*Resource Streams*
Defining the resource streams, in this case natural gas to the furnace

```{rosmose Furnace_rs}
: OSMOSE RESOURCE_STREAMS Furnace

|layer      |direction|value                |
|:----------|:--------|:--------------------|
|NATGAS     | in      |%Q_demand%           |
```

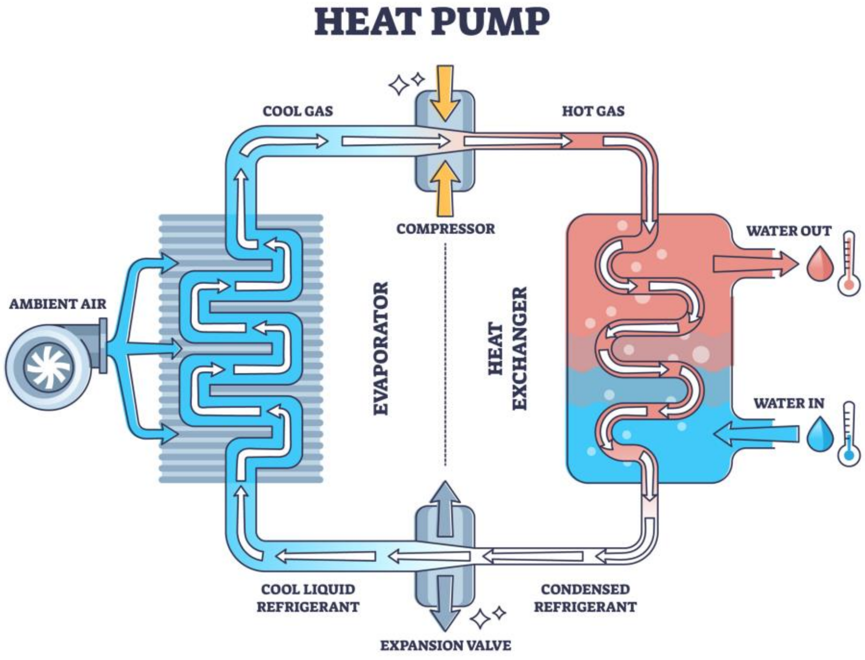

*Heat Streams*

```{rosmose Furnace_hs}
: OSMOSE HEAT_STREAMS Furnace

|name       |Tin   |Tout    |Hin             |Hout                |DT min/2             |alpha|
|:----------|:-----|:-------|:---------------|:-------------------|:--------------------|:----|
|radiation  |%Trad%|%Trad%  |%Q_radpreh%     |0                   |%dtmin_radiation%    |1    |
|convection |%Trad%|%Tstack%|%Q_conv_gross%  |0                   |%dtmin_convection%   |1    |
|preheating |%To%  |%Tprin% |0               |%Q_preh%            |%dtmin_convection%   |1    |
```
```

# Heat pump

- Using work to transfer heat from a cool space to a warm space using a refrigeration cycle
- 5 heat pumps available in the folder
- Modelled similarly




https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.powerpal.net%2Fblog%2Fheat-pumps%2F&psig=AOvVaw1BnurqG-mcRivslCSRquDC&ust=1706888770968000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCMin4aq-ioQDFQAAAAAdAAAAABAI

# Heat pumps – Rosmose ET

Electricity → Heat pump → Heat

- Inputs

```
# Heat Pump {-}

```{rosmose heatpump-et}
! OSMOSE ET heatpump
```


```{rosmose}
Evap_Tin = 54.5 [C] # Evaporator temperature inlet
Evap_Tout = 54.5 [C] # Evaporator temperature outlet
Cond_Tin = 76.5 [C] # Condenser temperature inlet
Cond_Tout = 76.5 [C] # Condenser temperature outlet
Cond_Qmax = 6000 [kW] # Condenser reference heat flow rate
exeff = 0.5 [-] # Second law efficiency
dtmin_2ph = 2 [C] # phase-change delta t minimum
n = 40.0 [yr] #lifetime of a heat pump
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

Note: Heat pumping will be profitable only if it allows to transfer heat from below to above the pinch, i.e.,
 if it transforms excess heat from the system heat source into useful heat for the system heat sink.

```{rosmose}
COPcarnot = (%Cond_Tin% + 273) / (%Cond_Tin% - %Evap_Tin%) [-] # Carnot COP
COP = %exeff% * %COPcarnot% [-] # Actual COP
W_heatpump = %Cond_Qmax% / %COP% [kW] # Heat pump power consumption
Evap_Qmax = %Cond_Qmax% * (%COP% - 1) / %COP% [kW] # Evaporator heat flow rate
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
Cinv2_HP = 300*%Cond_Qmax%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y] #300 Euro/kWth at the condenser
```
```

# Heat pumps – Rosmose ET

Electricity → Heat pump → Heat

- Layers and parameters

```
**Layers of the Heat Pump ET**

```{rosmose}
: OSMOSE LAYERS heatpump

|Layer      |Display name   |shortname  |Unit       |Color  |
|:----------|:--------------|:----------|:----------|:------|
|ELEC       |Electricity    |elec       |kW         |yellow |
```

**Heat Pump unit of the Heat pump ET**

```{rosmose}
: OSMOSE UNIT heatpump

|unit name  |type     |
|:----------|:--------|
|HeatPump   |Utility  |
```

**Parameters of the Heat Pump unit**

```{rosmose heatpump_params}
: OSMOSE UNIT_PARAM HeatPump

|cost1  |cost2  |cinv1  |cinv2      |imp1   |imp2   |fmin   |fmax   |
|:------|:------|:------|:----------|:------|:------|:------|:------|
|0      |0      |0      |%Cinv2_HP% |0      |0      |0      |10     |
```
```

# Heat pumps – Rosmose ET

Electricity → Heat pump → Heat

- Streams

```
*Resource Streams*
Defining the resource streams, in this case electricity to the heatpump

```{rosmose HeatPump_rs}        soline, 2 months ago • initialize book
: OSMOSE RESOURCE_STREAMS HeatPump

|layer      |direction|value              |
|:----------|:--------|:------------------|
|ELEC       | in      |%W_heatpump%       |
```

*Heat Streams*

```{rosmose HeatPump_hs}
: OSMOSE HEAT_STREAMS HeatPump

|name          |Tin        |Tout       |Hin        |Hout       |DT min/2   |alpha|
|:-------------|:----------|:----------|:----------|:----------|:----------|:----|
|evaporation   |%Evap_Tin% |%Evap_Tout%|0          |%Evap_Qmax%|%dtmin_2ph%|1    |
|condensation  |%Cond_Tin% |%Cond_Tout%|%Cond_Qmax%|0          |%dtmin_2ph%|1    |
```
```

# Market

The Market is modelling the transfers with the exterior, here.:

- the natural gas

- the electricity

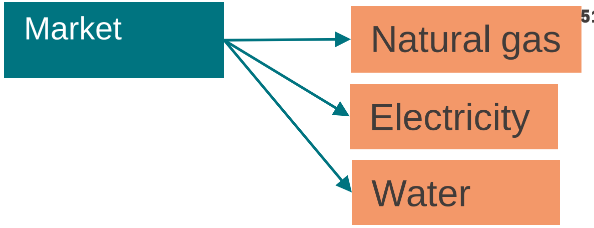
- the water

that are needed for the different units.

The costs associated are also defined in the Market.


https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.powerpal.net%2Fblog%2Fheat-pumps%2F&psig=AOvVaw1BnurqG-mcRivslCSRquDC&ust=1706888770968000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCMin4aq-ioQDFQAAAAAdAAAAABAI

# Market – Rosmose ET




- Inputs

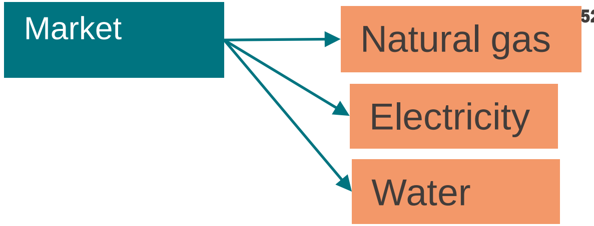
```
```{rosmose market}
! OSMOSE ET market
```

```{rosmose}
water_cost = 0.0025 [Euro/kg] # Water price in Switzerland is 2-2.5 CHF/m3 (wfw.ch and Swiss gas and water industry association)
CW_ref_LOAD = 1000 [kg/h] # Reference capacity of water supply
elec_cost = 0.25 [Euro/kWh] # price of electricity for businesses in Switzerland 2023 (Oiken)
ELEC_ref_POWER = 1000 [kW] # Reference capacity of electricity supply
natgas_cost = 0.119 [Euro/kWh] # price of natural gas for businesses in Switzerland (globalpetrolprices.com)
NATGAS_ref_LOAD = 1000 [kW] #Reference capacity of natural gas supply

```

```{rosmose}
CW_COST = %water_cost% * %CW_ref_LOAD% [Euro/h] # Reference cost of water supply
ELEC_SELL_COST = %elec_cost% * %ELEC_ref_POWER% [Euro/h] # Reference cost of electricity supply
NATGAS_COST = %natgas_cost% * %NATGAS_ref_LOAD% [Euro/h] # Reference cost of natural gas supply
```
```

# Market – Rosmose ET




- Layers and units

```
**Layers of the Market ET**

```{rosmose}
: OSMOSE LAYERS market

|Layer      |Display name   |shortname  |Unit       |Color  |
|:----------|:--------------|:----------|:----------|:------|
|NATGAS     |Gas            |ng         |kW         |green  |
|ELEC       |Electricity    |elec       |kW         |yellow |
|WATER      |Water          |water      |kg/h       |blue   |

```

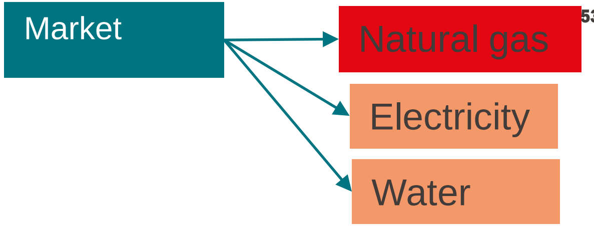
**Units of the Market ET**

```{rosmose}
: OSMOSE UNIT market

|unit name    |type   |
|:------------|:------|
|ElecSell     |Utility|
|NatgasSell   |Utility|
|WaterSell    |Utility|

```
```

# Market – Rosmose ET






- Natural gas unit

```
## Natural Gas Selling Unit {-}

**Parameters of the Natural Gas Selling unit**

```{rosmose NatgasSell_params}
: OSMOSE UNIT_PARAM NatgasSell

|cost1  |cost2          |cinv1  |cinv2  |imp1   |imp2   |fmin   |fmax   |
|:------|:--------------|:------|:------|:------|:------|:------|:------|
|0      |%NATGAS_COST%  |0      |0      |0      |0      |0      |1000   |
```

**Natural Gas Selling Streams**

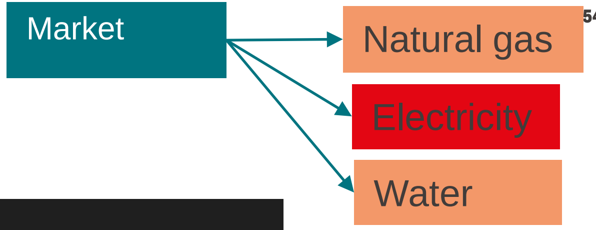
*Resource Streams*

Natural gas sold from the market to the process. In addition to total CO~2~ emissions (direct and indirect) from the use of natural gas

```{rosmose NatgasSell_rs}
: OSMOSE RESOURCE_STREAMS NatgasSell

|layer     |direction|value                |
|:---------|:--------|:--------------------|
|NATGAS    |out      |%NATGAS_ref_LOAD%    |
```
```

# Market – Rosmose ET




- Electricity unit

```
## Electricity Selling Unit {-}

Electricity sold by the grid to the process

**Parameters of the Electricity Selling unit**

```{rosmose ElecSell_params}
: OSMOSE UNIT_PARAM ElecSell

|cost1  |cost2              |cinv1  |cinv2  |imp1   |imp2   |fmin   |fmax   |
|:------|:------------------|:------|:------|:------|:------|:------|:------|
|0      |%ELEC_SELL_COST%   |0      |0      |0      |0      |0      |100000 |
```

**Electricity Selling Streams**

*Resource Streams*

Electricity sold from the market to the process and the indirect CO~2~ emissions from the electricity generated by the grid.

```{rosmose ElecSell_rs}
: OSMOSE RESOURCE_STREAMS ElecSell

|layer     |direction|value              |
|:---------|:--------|:------------------|
|ELEC      | out     |%ELEC_ref_POWER%   |
```
```

EPFL

# Market – Rosmose ET

- Water unit

```
## Water Selling Unit {-}
Water from the market to the process

**Parameters of the Water Selling unit**

```{rosmose WaterSell_params}
: OSMOSE UNIT_PARAM WaterSell

|cost1  |cost2      |cinv1  |cinv2  |imp1   |imp2   |fmin   |fmax   |
|:------|:----------|:------|:------|:------|:------|:------|:------|
|0      |%CW_COST%  |0      |0      |0      |0      |0      |10000  |
```

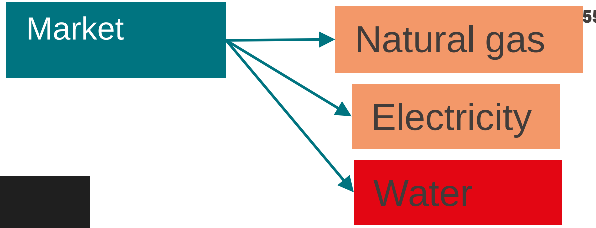
**Water Selling Streams**

*Resource Streams*

Water sold from the market to the process

```{rosmose WaterSell_rs}
: OSMOSE RESOURCE_STREAMS WaterSell

|layer|direction|value         |
|:----|:--------|:-------------|
|WATER|out      |%CW_ref_LOAD%  |
```
```

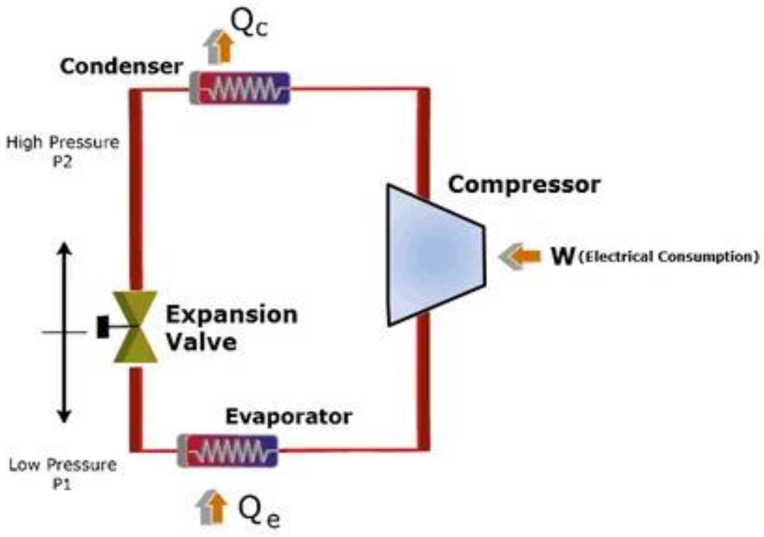


# Refrigerator



- **This unit removes unwanted heat**

1. A **compressor** pressurizes the refrigerant.
2. A **condenser**, where the refrigerant condenses from vapor to its liquid form giving off heat.
3. A **metering device** regulates the flow and consequently lowers the pressure, of the refrigerant.
4. An **evaporator** also referred to as a cooling coil, where the refrigerant expands removing heat from the area as the refrigerant "evaporates" changing into its vapor state once again.




https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.araner.com%2Fblog%2Findustrial-refrigeration-cycle-components&psig=AOvVaw164yOgD1a4cO07oBXuB-N1&ust=1706890908898000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCPimoKbGioQDFQAAAAAdAAAAABAD

EPFL

# Refrigerator – Rosmose ET

Electricity → Refrigerator → Cold

- Inputs

```
# Refrigerator {-}

```{rosmose refrigerator}
: OSMOSE ET refrigerator
```

```{rosmose}
Evap_Tin = 5 [C] # Evaporator temperature inlet
Evap_Tout = 5 [C] # Evaporator temperature outlet
Cond_Tin = 35 [C] # Condenser temperature inlet
Cond_Tout = 35 [C] # Condenser temperature outlet
Evap_Qmax = 5000 [kW] # evaporator reference heat flow rate (Q_L)
exeff = 0.5 [-] # Second law efficiency
dtmin_2ph = 2 [C] # phase-change delta t minimum
n = 40.0 [yr] #lifetime
i = 0.06 [-] #interest rate
CEPCI_2020 = 596.2 [-] # actual CEPCI
CEPCI_2008 = 575.4 [-] # CEPCI 2008
```

```{rosmose}
COPcarnot = (%Evap_Tin% + 273) / (%Cond_Tin% - %Evap_Tin%) [-] # Carnot COP
COP = %exeff% * %COPcarnot% [-] # Actual COP
W_refrig = %Evap_Qmax% / %COP% [kW] # Heat pump power consumption
Cond_Qmax = %Evap_Qmax% * (%COP% + 1) / %COP% [kW] # Condenser heat flow rate (Q_H)
Annuity = (%i%*(1+%i%)**%n%)/((1+%i%)**%n%-1) [-] #annualization factor
Cinv2_RF = 300*%Cond_Qmax%*(%CEPCI_2020%/%CEPCI_2008%)*%Annuity% [Euro/y] #300 Euro/kWth at the condenser
```
```

# Refrigerator – Rosmose ET

Electricity → Refrigerator → Cold



- Layers, units, parameters

```
**Layers of the Refrigerator ET**

```{rosmose}
: OSMOSE LAYERS refrigerator

|Layer      |Display name|shortname|Unit    |Color  |
|:----------|:-----------|:--------|:-------|:------|
|ELEC       |Electricity |elec     |kW      |yellow |
```

**Refrigerator unit of the Refrigerator ET**

```{rosmose}
: OSMOSE UNIT refrigerator

|unit name      |type   |
|:--------------|:------|
|Refrigerator   |Utility|
```

**Parameters of the Refrigerator unit**

```{rosmose Refrigerator_params}
: OSMOSE UNIT_PARAM Refrigerator

|cost1  |cost2  |cinv1  |cinv2      |imp1   |imp2   |fmin   |fmax   |
|:------|:------|:------|:----------|:------|:------|:------|:------|
|0      |0      |0      |%Cinv2_RF% |0      |0      |0      |10     |
```
```

# Refrigerator – Rosmose ET

Electricity → Refrigerator → Cold

- Streams

```
**Refrigerator Streams**

*Resource Streams*
Defining the resource streams, in this case electricity to the refrigerator

```{rosmose Refrigerator_rs}
: OSMOSE RESOURCE_STREAMS Refrigerator

|layer      |direction|value              |
|:---------|:-------|:------------------|
|ELEC       | in      |%W_refrig%          |
```

*Heat Streams*

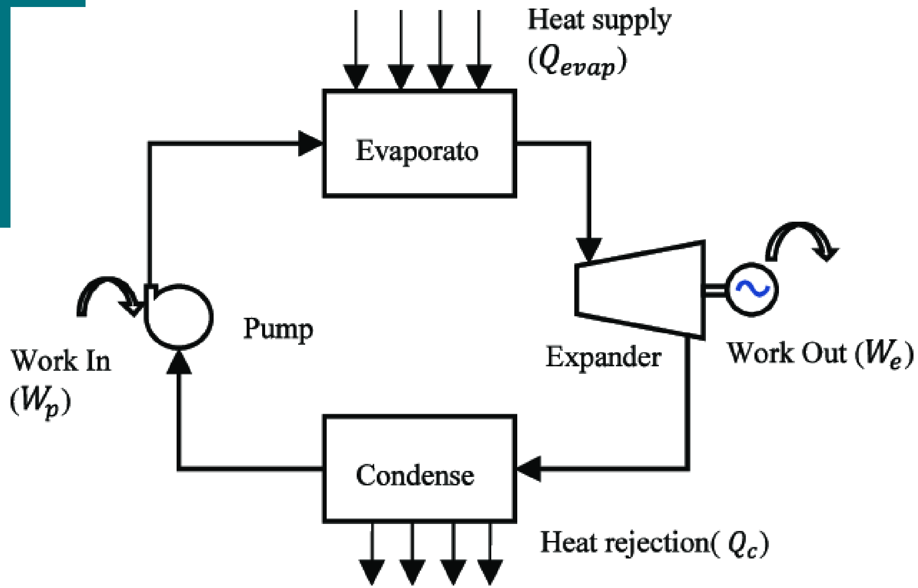
```{rosmose Refrigerator_hs}
: OSMOSE HEAT_STREAMS Refrigerator

|name          |Tin        |Tout       |Hin        |Hout       |DT min/2   |alpha|
|:------------|:---------|:---------|:---------|:---------|:---------|:----|
|evaporation   |%Evap_Tin% |%Evap_Tout%|0          |%Evap_Qmax%|%dtmin_2ph%|1    |
|condensation  |%Cond_Tin% |%Cond_Tout%|%Cond_Qmax%|0          |%dtmin_2ph%|1    |
```
```

# Superstructures:

## Rankine Cycle and Heat pump

- It is a thermodynamic cycle converting thermal heat into mechanical work or vice versa, using a fluid called refrigerant.



EPFL

IPESE
Industrial Process and
Energy Systems Engineering


# Thank you!

École
polytechnique
fédérale
de Lausanne