

CLEWs

Hands-on lecture 3: Creating a simple energy model with OSeMOSYS

Useful links:

- 1) Link to open [Momani](#) in the local computer
- 2) [Discussion forum](#) for OSeMOSYS
- 3) [Results from this Hands-on](#)

Learning outcomes

By the end of this exercise you will be able to:

- 1) Create technologies and commodities related to thermal power
- 2) Input key numerical data for the technologies and commodities
- 3) Run a simple model and observe the results

Overview

The figure gives an overview of the CLEWs model you will create by the end of all the hands-on sessions. You will build it little by little starting with this Hands-on session. The figure shows a 'Reference Energy system' (RES), but also includes the land and water systems. As discussed in Lecture 2, the boxes represent technologies, processes or physical assets. The lines represent flows of commodities.

The summary of the naming convention is given in the table below. You do not have to memorise it, you will be gradually introduced to it with every hands-on session. However, you can always come back to this table if you are uncertain. It includes all technologies and commodities that you will encounter through all hands-on sessions.

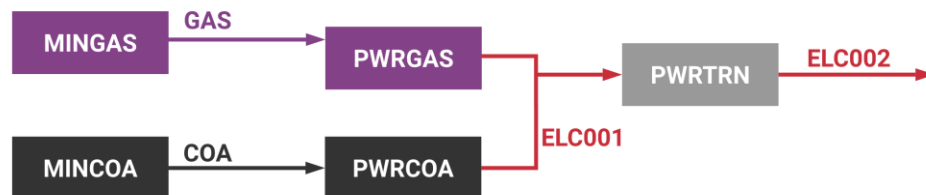
Name	Description
<i>Technologies</i>	
MINLND	Land resource
LNDAGR001	Land for agriculture and other uses
LNDMAIHR	Land representing rainfed maize cultivation
LNDRICH	Land representing rainfed rice cultivation
LNDMAIHI	Land representing irrigated maize cultivation
LNDRICHI	Land representing irrigated rice cultivation
LNDFOR	Land representing forests
LNDBLT	Land representing built-up land
LNDWAT	Land representing water bodies
MINPRC	Precipitation water resource
MINGAS	Gas extraction
MINCOA	Coal extraction
MINHYD	Hydro resource for power
MINSOL	Solar resource for power
MINWND	Wind resource for power
PWRGAS	Gas power plant

PWRCOA	Coal power plant
PWRHYD	Hydro power plant
PWRSOL	Rooftop solar photovoltaic
PWRWND	Wind turbines
PWRTRN	Transmission and distribution network
DEMAGRDSL	Diesel used in the agriculture sector
DEMAGRSURWAT	Surface water supply for agriculture
DEMAGRGWTWAT	Ground water supply for agriculture
DEMPWRSURWAT	Surface water supply for power plants
DEMPWRGWTWAT	Ground water supply for power plants
DEMPUBSURWAT	Surface water supply for public use
DEMPUBGWTWAT	Ground water supply for public use
DEMTRABIO	Biofuel for transport
Commodities	
LND	Land
CRPMAI	Maize
CRPRIC	Rice
WTRPRC	Precipitation water
AGRWAT	Agricultural water
WTREVT	Evapotranspired water
WTRGWT	Ground water
WTRSUR	Surface water
PWRWAT	Water for power plants
PUBWAT	Public water
GAS	Natural gas
COA	Coal
HYD	Hydro
SOL	Solar
WND	Wind
ELC001	Electricity fed to transmission and distribution network
ELC002	Electricity for final uses
AGRDSL	Diesel used in the agriculture sector
LMAIHR	Land representing rainfed maize cultivation
LRICHR	Land representing rainfed rice cultivation
LMAIHI	Land representing irrigated maize cultivation
LRICHI	Land representing irrigated rice cultivation
LFOR	Land representing forests
LBLT	Land representing built-up land
LWAT	Land representing water bodies
TRABIO	Biofuel for transport

Activity 1 – Introduce new technologies and commodities

The starting point for this activity will be the template model you obtained when installing MoManI. It is called **workshop_model_v0**. If you carried out correctly Hands-on 1, you will find it by opening MoManI on your browser: <http://localhost:8080/#/models>.

In this activity, you will create the first small part of the RES shown in the figure above. The part you create is one part of the electricity supply chain and it includes a demand, the transmission and distribution network, two types of power plants and the primary supply of fuels for those power plants. This is shown in the figure below. In the coming hands-on sessions, you will add more pieces to this.



To create the above, you will add technologies and commodities to the SETs of your model and assign the first numerical parameters for each of them.

You will have to add 5 technologies (the 5 boxes in the figure above):

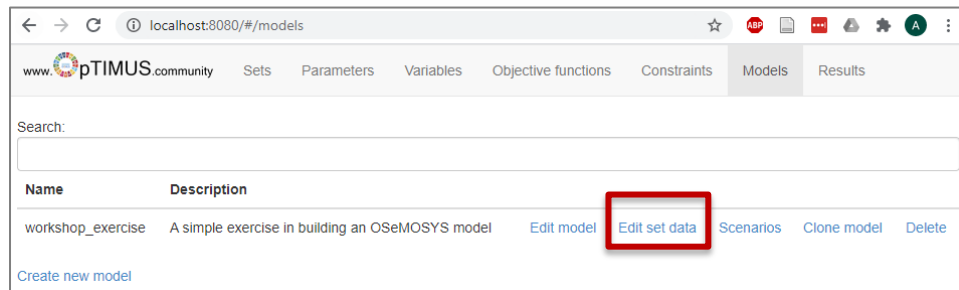
- **MINGAS** (supply of gas for the power sector)
- **MINCOA** (supply of coal for the power sector)
- **PWRGAS** (gas-fired power plant)
- **PWRCOA** (coal-fired power plant)
- **PWRTRN** (electricity transmission and distribution network)

And 4 commodities:

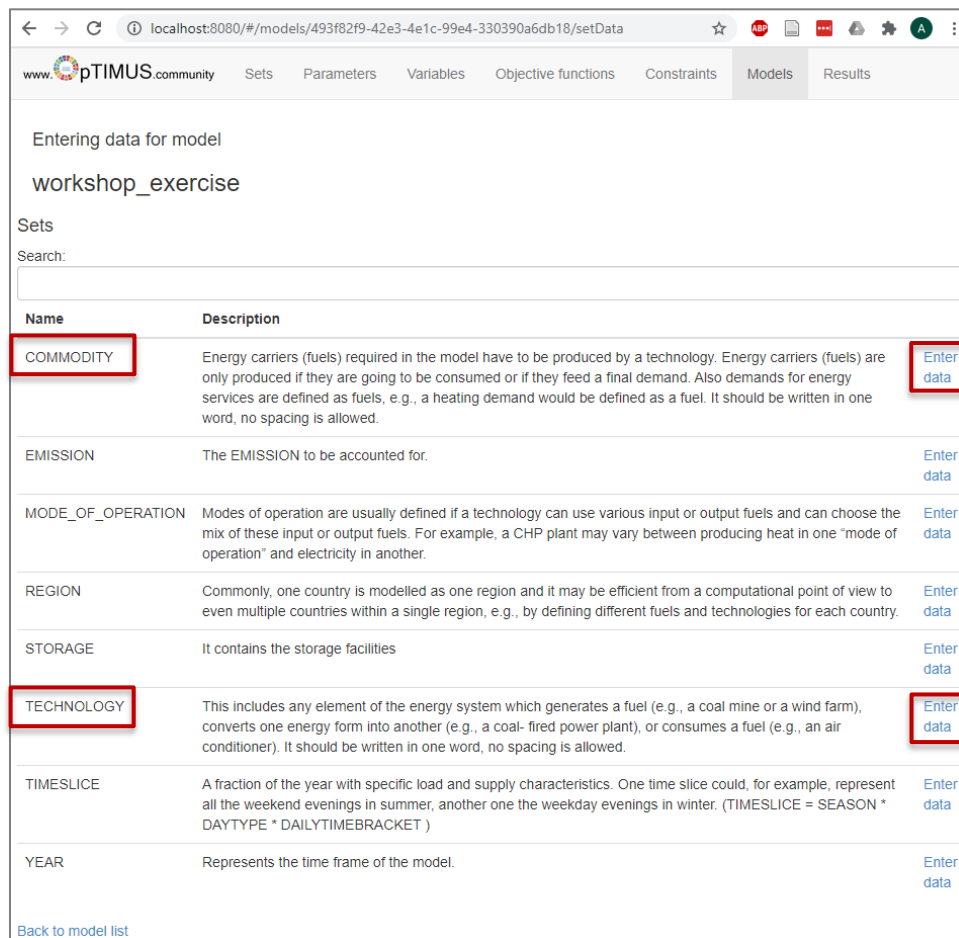
- **GAS** (Natural gas)
- **COA** (Coal)
- **ELC001** (Electricity fed to the transmission and distribution network)
- **ELC002** (Final electricity, delivered by the transmission and distribution network)

For creating technologies and commodities, in the MoManI page:

1. Click in the top bar on **Models**
2. Then, click on **Edit set data**



3. Find the set you want to edit (i.e. **TECHNOLOGY** for adding the technologies and **COMMODITY** for adding the commodities)
4. Click **Enter data**



- Click '+' then enter the name (**exactly as indicated above, with no spaces in it, nor special characters – any deviation from the names indicated above may make the model crash and/or prevent you from visualizing results later on**). You do not need to enter anything in the 'Description' column.



www.pTIMUS.community Sets Parameters Variables Objective functions Constraints Models Results

Data entry for set TECHNOLOGY

Name	Description	Group	Color	Remove
+				

Groups

Name

+

Save Cancel



www.pTIMUS.community Sets Parameters Variables Objective functions Constraints Models Results

Data entry for set COMMODITY

Name	Description	Group	Color	Remove
+				

Groups

Name

+

Save Cancel

- Click **Save** (**always do it, at the end of any step, with any element you modify!**)

With these steps, you have introduced in the model new technologies and new commodities. Now you will need to create links between them. I.e. you will define which commodities flow in and out of which technologies (and with what efficiency), in order to create an energy supply chain. The supply chain you will create is shown again in the figure below, for your convenience.



This chain indicates that (starting from the right): there is a demand for electricity (ELC002); this electricity is supplied to the consumers through the transmission and distribution network (PWRTRN); gas power plants (PWRGAS) and coal power plants (PWRCOA) are the two technologies supplying electricity to the network (for now); for supplying the electricity, the gas power plants are fed with natural gas (GAS) and the coal power plants with coal (COA); gas and coal come from extraction activities, respectively represented by MINGAS and MINCOA.

N.B. In OSeMOSYS the supply chain must always start with a technology. I.e. there must be a technology making the needed commodity available. In this case, at the start of the supply chain there are two technologies representing extraction of gas and of coal. **Failing to start a supply chain with a technology will make that supply chain not work and the model, potentially, crash.**

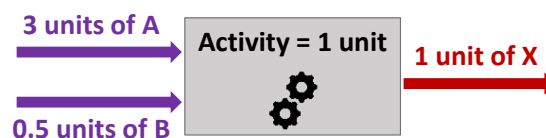
In order to define the links between technologies and commodities, it is useful to recall some concepts:

- **Activity:** it refers to any process occurring within a technology (e.g. fuel combustion, water treatment, crude oil refining, crop harvesting)
- **InputActivityRatio:** ratio between an input commodity and technology activity
- **OutputActivityRatio:** ratio between an output commodity and technology activity

Examples follow.

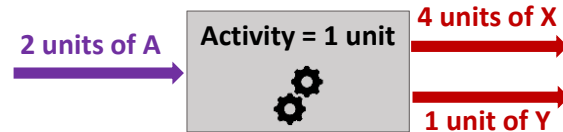
Example 1: Multiple inputs, single output

- 1 unit of activity requires 3 units of commodity A and 0.5 units of commodity B
- 1 unit of activity produces 1 unit of X



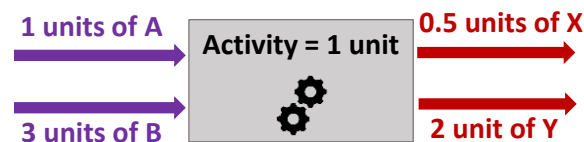
Example 2: Single input, multiple outputs

- 1 unit of activity requires 2 units of commodity A
- 1 unit of activity produces 4 units of X and 1 unit of Y

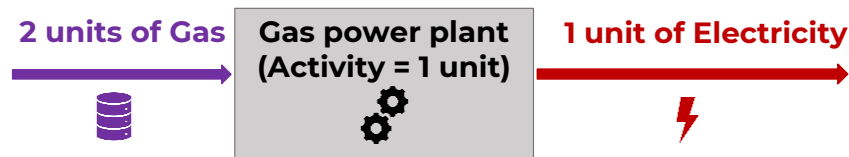


Example 3: Multiple inputs, multiple outputs

- 1 unit of activity requires 1 unit of commodity A and 3 units of commodity B
- 1 unit of activity produces 0.5 units of X and 2 unit of Y



N.B. When defining the InputActivityRatio and the OutputActivityRatio, **you automatically define also the efficiency of the technology**. See the example below:



The efficiency of the process of electricity generation is the ratio between the output and the input. In this case:

$$Efficiency = \frac{1 \text{ unit of electricity}}{2 \text{ units of Gas}} = 50\%$$

If the output and the input have the same type of unit (e.g. PJ), the efficiency will be a ratio with no unit. If the output and the input have different types of units, then the efficiency will have units. The choice of the units lies with you. In MoManI you just enter numbers, you do not enter the units. Therefore, you must keep a separate account of what units the numbers are meant to have. In the hands-ons for this course, we will always indicate the exact values you need to input and in which units they are intended.

Continuing the exercise, you are now asked to introduce the InputActivityRatio and OutputActivityRatio for PWRGAS and PWRCOA so that they both produce 1 PJ of electricity, with efficiency respectively of 50% and 33%. You will also enter data for the other technologies, so that they all have efficiency 100%.

Start from the **OutputActivityRatio**:

1. Click **Models**

2. Choose your model from the list
3. Click **Scenarios**
4. Click **Enter data**
5. Search for the parameter **OutputActivityRatio** and click **Enter data (slices)**
6. Fix REGION and MODE_OF_OPERATION. Then fix one COMMODITY at a time. A data table will appear for each with all the technologies in the model listed **for all years**. Enter data for each.

Data entry for parameter OutputActivityRatio

Default value

Fix dimensions

REGION ▼ EXAMPLE (Example region) ▼

MODE_OF_OPERATION ▼ 1 ▼

COMMODITY ▼ COA ▼

Switch axes

	2019	2020	2021	2022
MINCOA				
MINGAS				
PWRCOA				
PWRGAS				
PWRTRN				

Other parameters

7. You will have to enter the data given in the table below, **for all years**. The table below shows which technology you give the value for, which commodity you need to fix for that technology and which value needs to be given for each year. For instance, for MINCOA, you will need to fix the commodity COA and give the value 1 PJ/PJ (PJ output per PJ activity) for all years. By doing so, you will be defining that the technology MINCOA (coal extraction) provides 1 unit of COA (coal) for every unit of its activity. This corresponds to the figure above, where COA is drawn as an output of MINCOA
8. Click **Save** after entering the values for all the couples technology-commodity shown below. You do not need to save before then

Technology	Value to be given	Output commodity
MINGAS	1	GAS

MINCOA	1	COA
PWRGAS	1	ELC001
PWRCOA	1	ELC001
PWRTRN	1	ELC002

Note that we always prefer setting the output as 1. This means that, for efficiencies < 100%, the input will be a number higher than 1. The reason for setting the output as 1 is linked to the structure of OSeMOSYS and requires an advanced explanation. If you are curious, you may ask the question on the forum.

Now move to the **InputActivityRatio**:

- Click **Models**
- Choose your model from the list
- Click **Scenarios**
- Click **Enter data**
- Search for the parameter **InputActivityRatio** and click **Enter data (slices)**
- Fix REGION and MODE_OF_OPERATION. Then select one COMMODITY at a time. A data table will appear for each with all the technologies in the model listed **for all years**. Enter data for each
- You will have to enter the data given in the table below
- Click **Save**

Technology	Value to be given	Input commodity
PWRGAS	2	GAS
PWRCOA	3	COA
PWRTRN	1	ELC001

Note that the value 2 PJ/PJ for the gas power plant comes from 1/50% and the value 3 PJ/PJ for the coal power plant comes from 1/33%. That is, the efficiencies can be used to calculate the InputActivityRatios of power plants.

Activity 2 – Introduce a demand for electricity

Now you have set up the supply side of the energy chain. However, the power plants will supply electricity (and compete in the supply) only if there is demand for electricity. In this activity you will introduce a demand for electricity, so that, driven by the demand, the

model will try and find the best supply option(s) and will suggest how much the power plants should operate.

In OSeMOSYS, demands are user-defined (exogenous) and can be set for one or more commodities. They “drive” the optimisation process, that is, the objective of the model is to minimise the cost of meeting them.

The user can introduce two types of demand in OSeMOSYS:

- **AccumulatedAnnualDemand:** it must be balanced by the supply on an annual basis
- **SpecifiedAnnualDemand:** it must be met based on a “time-of-use” profile such as daily fluctuations in electricity demand.

You will learn more about the difference between the two in Lecture 5 and in the following hands-on sessions. For now, you will introduce values only for the AccumulatedAnnualDemand. In this way, you will define a value of demand for each year of the time domain. The supply chain needs to meet this demand over the year.

Introduce a value of 100 PJ as AccumulatedAnnualDemand for electricity ELC002 in every year, in the following way:

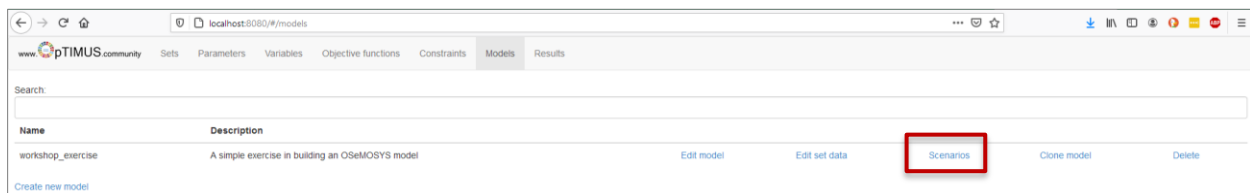
- Click **Models**
- Choose your model from the list
- Click **Scenarios**
- Click **Enter data**
- Search for the parameter **AccumulatedAnnualDemand** and click **Enter data (slices)**
- Fix the dimension ‘REGION’ and a data table will appear with all the commodities in the model listed. Enter data for ELC002 as suggested in the table below
- Click **Save**

Commodity	2019	2020	2021	2022
ELC002	100	100	100	100

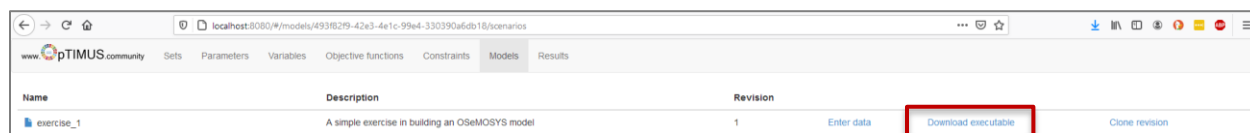
Activity 3 – Run the model and visualize the results

If you have carried out all the above steps correctly, you are now ready to run the optimisation and see which technology or technologies will be chosen to meet the electricity demand. At this stage, we have given very little input. Therefore, the optimisation will unlikely give very meaningful results. However, it is good to go through the process of running a model, checking if the run completes and understanding how to read the results. For running the model and visualizing the results, take the following steps:

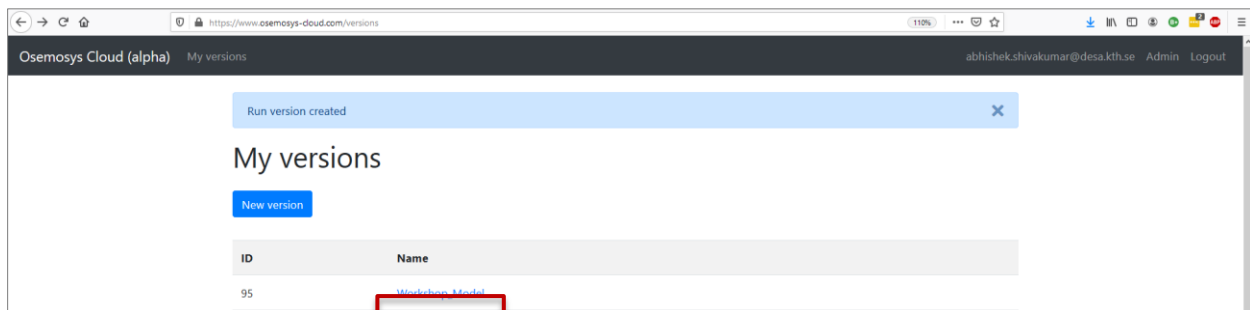
1. On MoManI, next to your model, click **Scenarios**



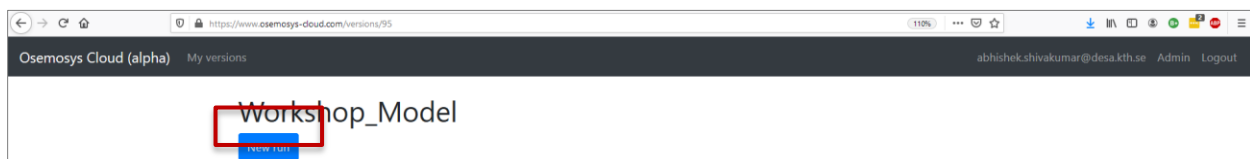
2. Click **Download executable**. This will download a zipped folder.



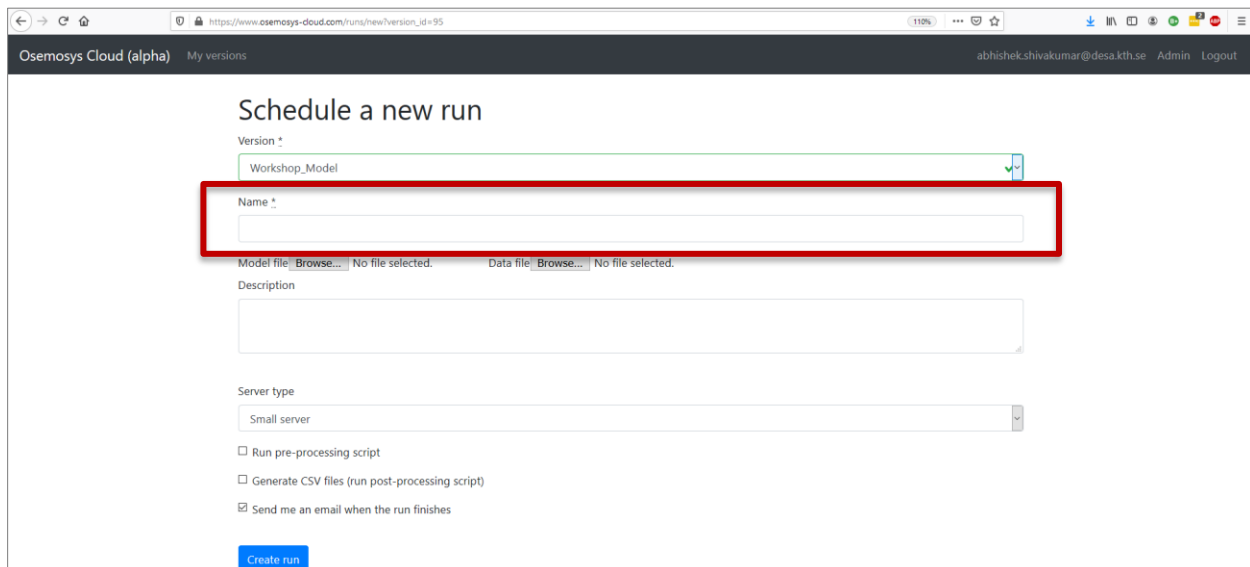
3. Go to the folder where this zipped folder has been just downloaded and unzip it. You may move the unzipped folder in any directory of convenience for you. Remember where you put it. The folder contains, among others, two elements of importance for you: a file called **data.txt**, with all the inputs you have provided so far; a file called **model.txt** which contains all the equations of OSeMOSYS, i.e. the linear optimisation problem. You need not do anything with them now, you will see their use in a moment.
4. Now, connect to <https://www.osemosys-cloud.com/> and sign in with your email and password
5. Click on the model and then the version you created in Hands-on 1 (we named it **Workshop_Model**)



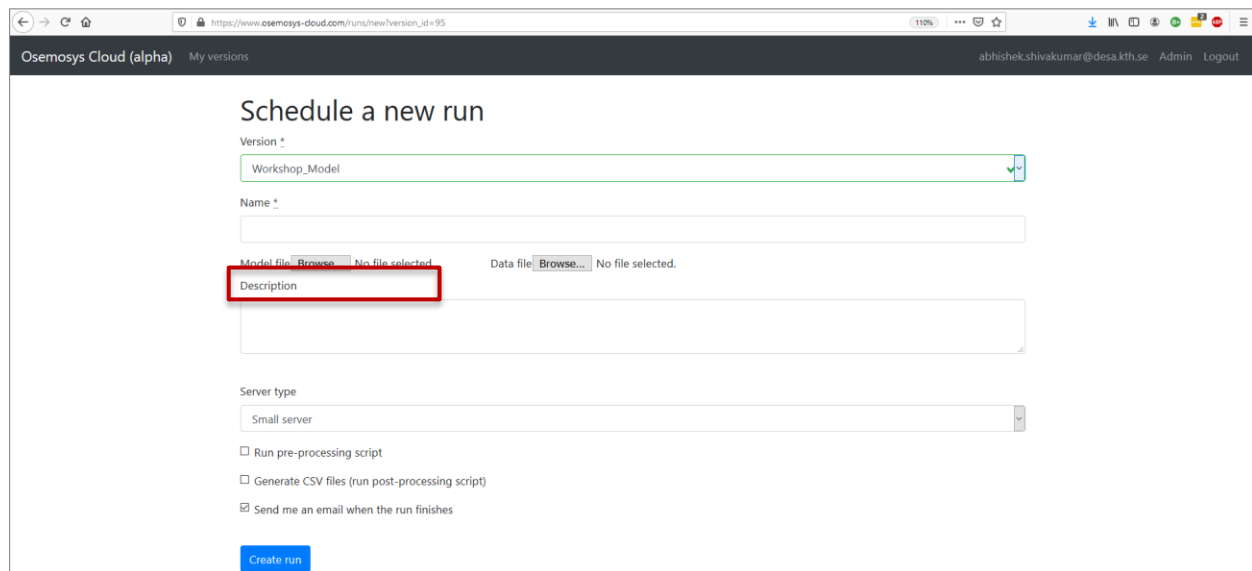
6. Click on **New run** to create a model run



7. Enter a name for your model run (do not use spaces or special characters!). You may also add a short description, but that is not necessary



8. Next to the writing 'Model file', click **browse** to look for and upload the **model.txt** file that is in the unzipped folder you downloaded a minute ago



Schedule a new run

Version *
Workshop_Model

Name *

Model file: [Browse...](#) No file selected. Data file: [Browse...](#) No file selected.

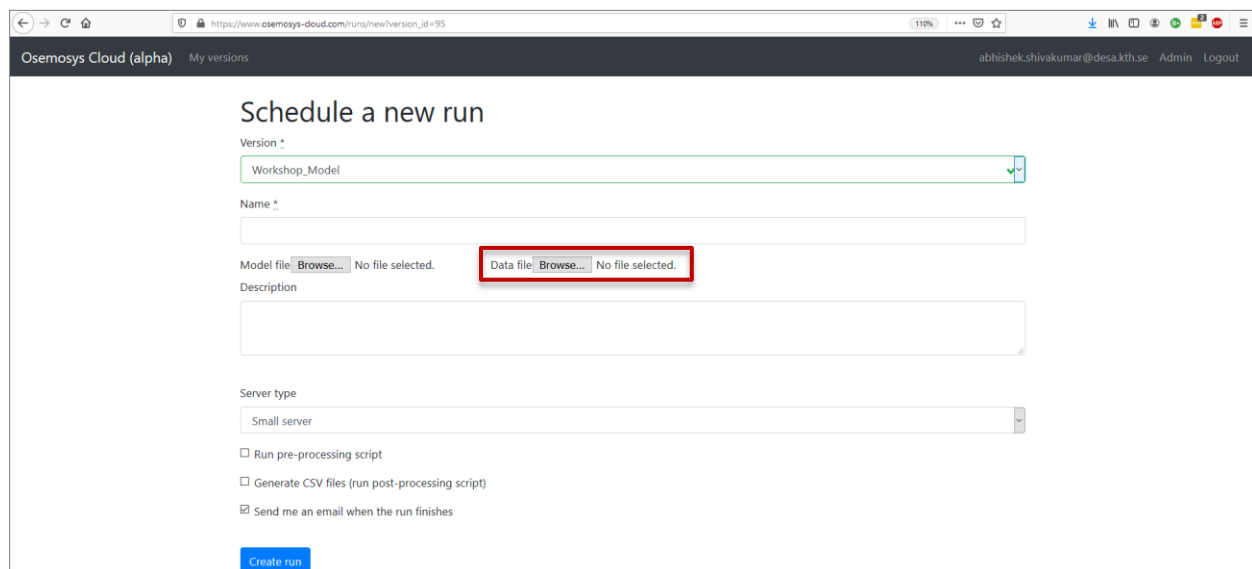
Description

Server type
Small server

☐ Run pre-processing script
☐ Generate CSV files (run post-processing script)
☒ Send me an email when the run finishes

[Create run](#)

9. Next to the writing 'Data file', click **browse** to look for and upload the **data.txt** file that is in the unzipped folder you downloaded a minute ago



Schedule a new run

Version *
Workshop_Model

Name *

Model file: [Browse...](#) No file selected. Data file: [Browse...](#) No file selected.

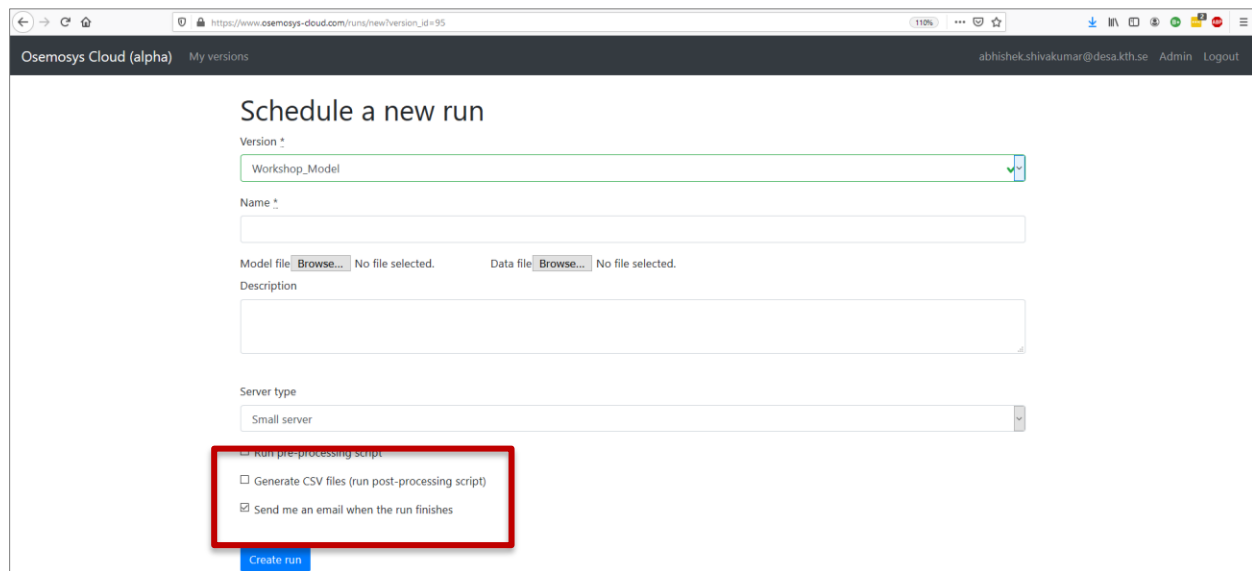
Description

Server type
Small server

☐ Run pre-processing script
☐ Generate CSV files (run post-processing script)
☒ Send me an email when the run finishes

[Create run](#)

10. Un-check the boxes '**Run pre-processing script**' and '**Generate CSV files**' at the bottom. You may also uncheck the box '**Send me an email when the run finishes**', if you do not want to receive email notifications when the run is done.



Schedule a new run

Version *
Workshop_Model

Name *

Model file: [Browse...](#) No file selected. Data file: [Browse...](#) No file selected.

Description

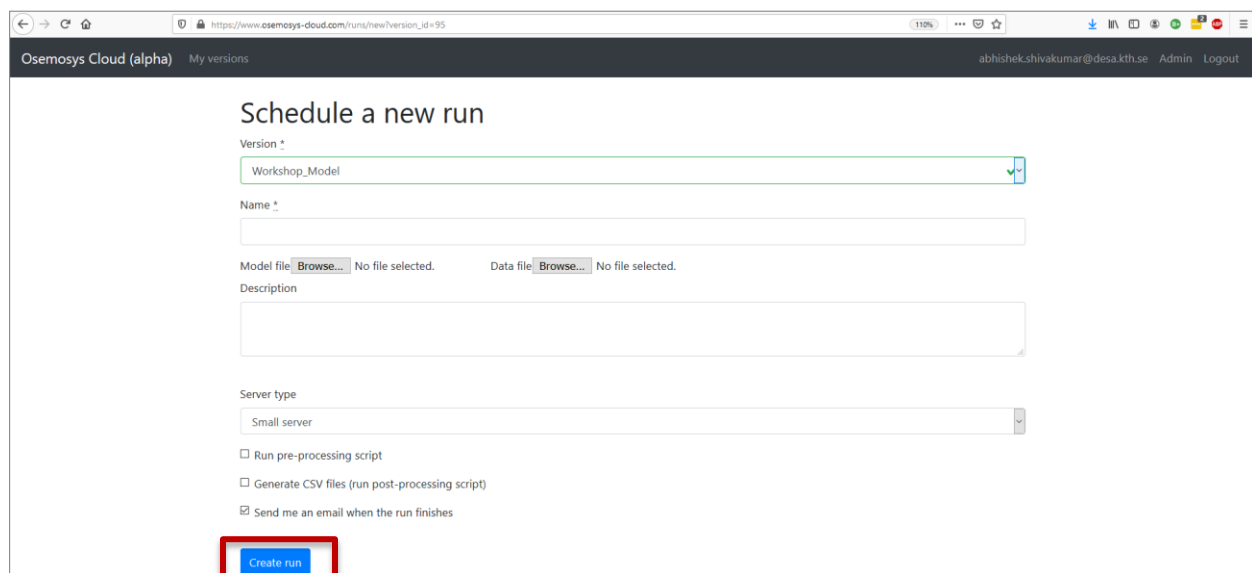
Server type
Small server

☐ Run pre-processing script
☐ Generate CSV files (run post-processing script)
☒ Send me an email when the run finishes

[Create run](#)

11. You will also see a drop down menu to choose the Server type. The default setting there will be **Small server**. That means that your model will be run on a small server. That is more than sufficient for this little model, so do not change the setting.

12. Click **Create run**



Schedule a new run

Version *
Workshop_Model

Name *

Model file: [Browse...](#) No file selected. Data file: [Browse...](#) No file selected.

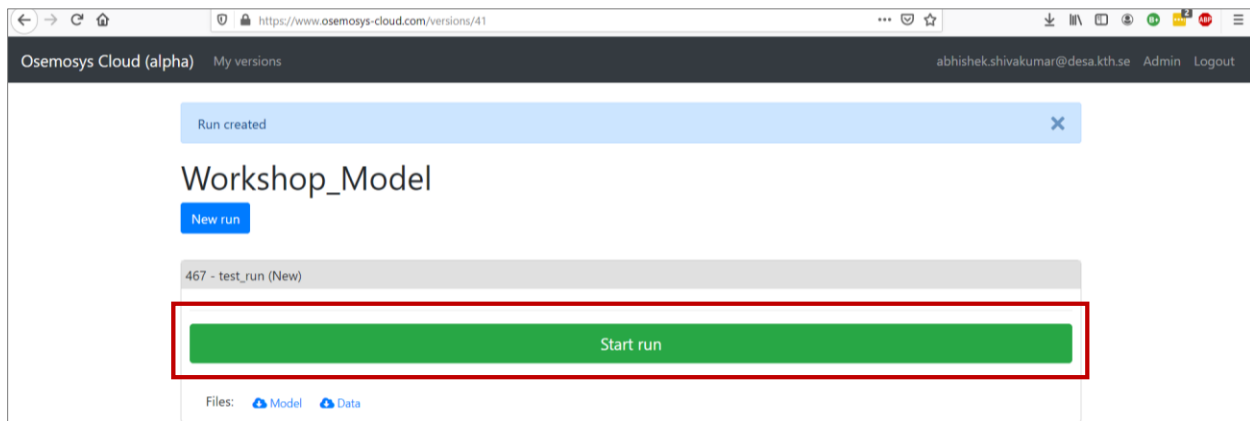
Description

Server type
Small server

☐ Run pre-processing script
☐ Generate CSV files (run post-processing script)
☒ Send me an email when the run finishes

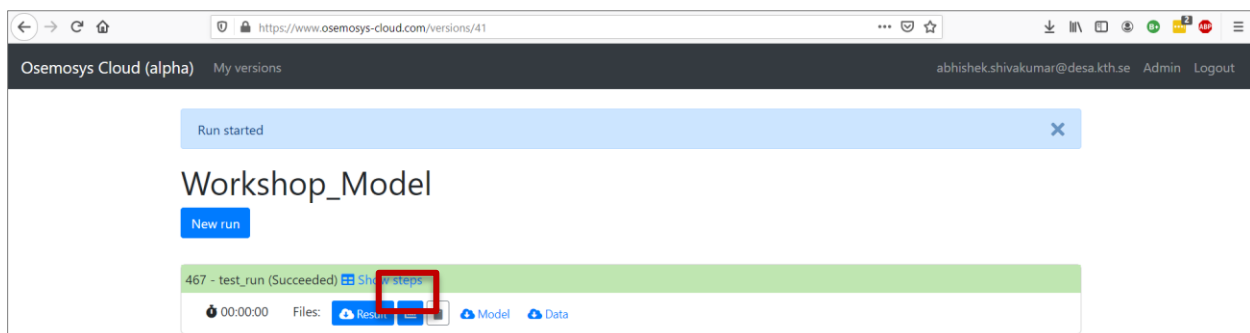
[Create run](#)

13. Click **Start run**



This will start the run, which should take a couple of seconds to complete. If the run is completed successfully, the bar of the run will be green and will show the message (Succeeded) next to the name of the run (see below). If after some time you do not see the below, you may need to refresh the page.

To view the results of this run, **click on the symbol with the little graph**.



What do you see? If there is any result shown, are you able to understand what is being shown? Answer the quiz for this hands-on!

N.B. You will have to repeat this operation for all the times you will be asked to run the model in the next hands-on sessions. Every time you will have to create a New run (we suggest you give them meaningful names, so that you are able to distinguish all steps from each other)