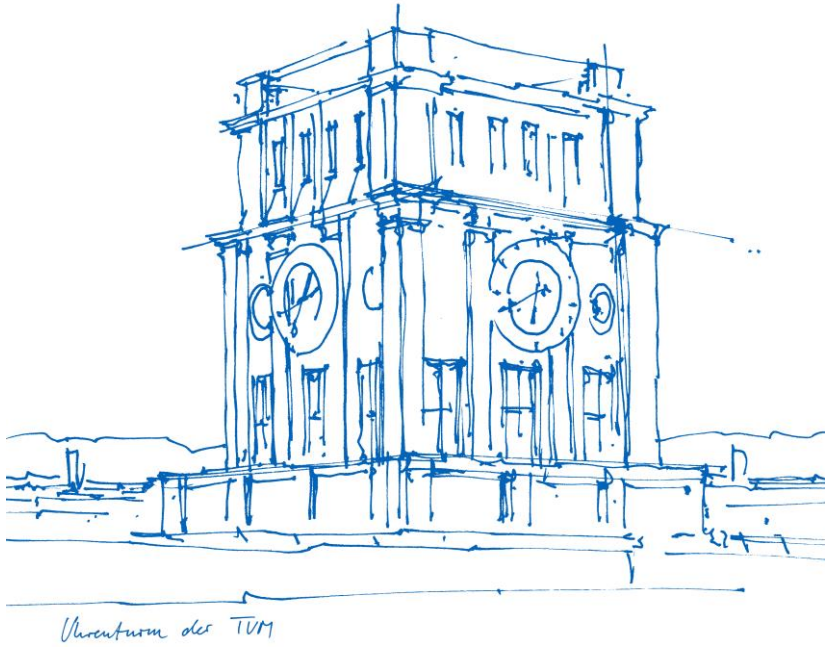
PropENS – Panda Power Project

Jiahe Chu, Karim Kaaniche, Martin Oviedo, Rohith Sureshbabu, Taeyoung Kim

Final Presentation Project Lab ENS

Garching, 10 Feburary 2023

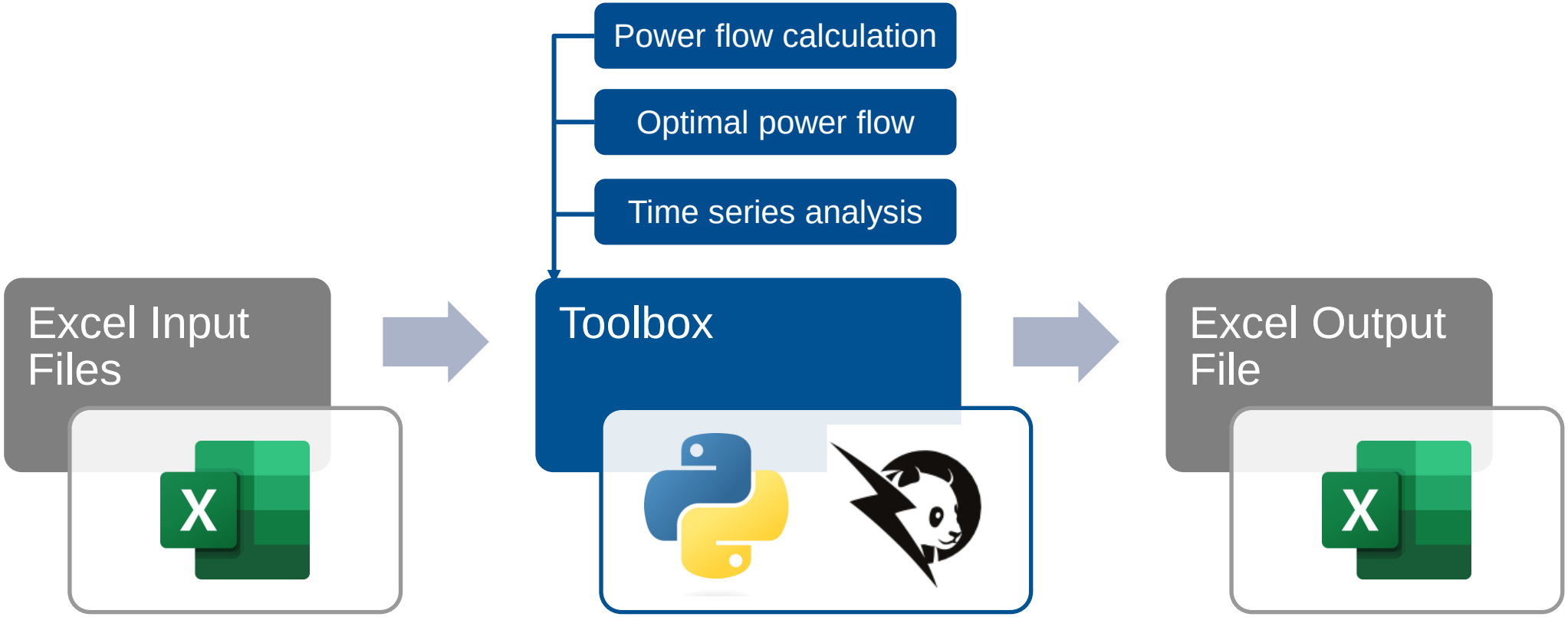


Overview

- Introduction
- Front-end
- Database
- Analysis
- Testing & Error Management

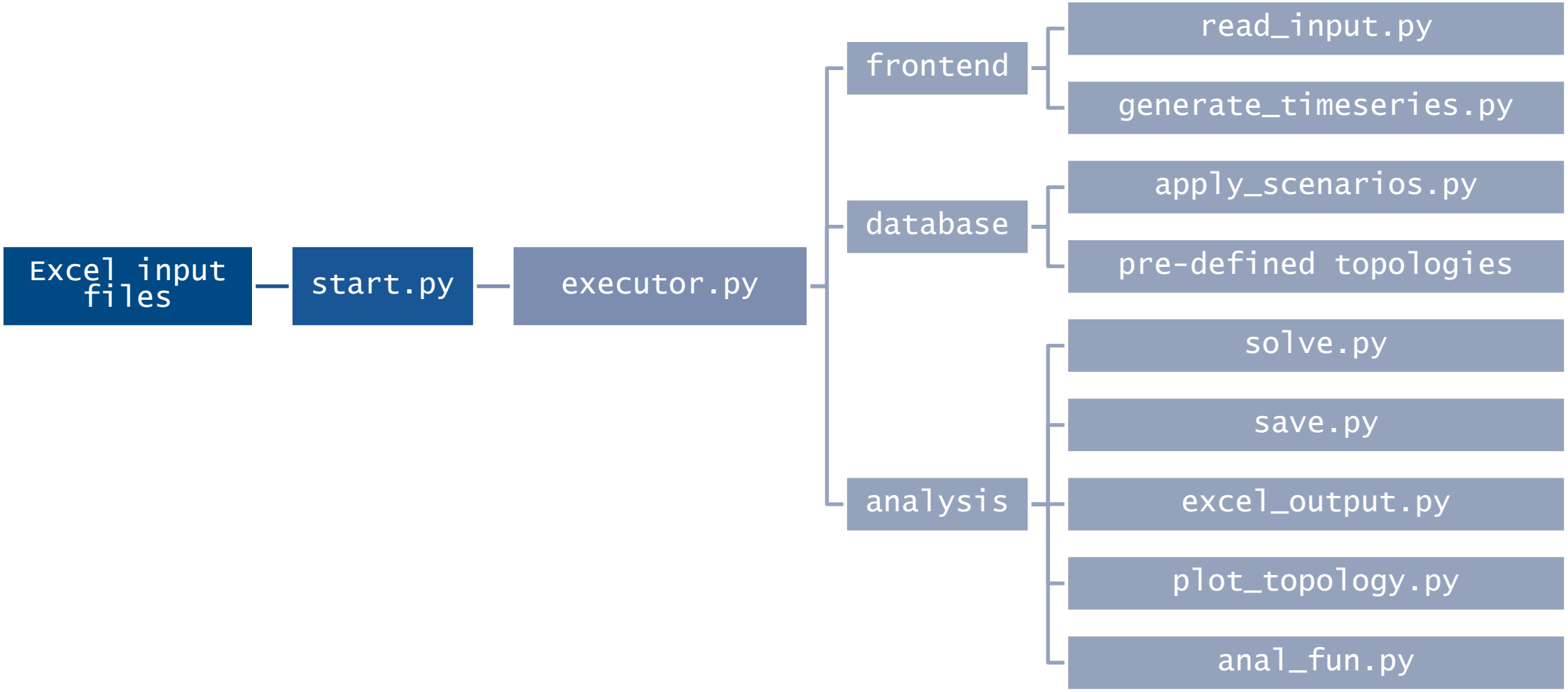
Introduction

Introduction: Project Overview



a toolbox using pandapower to do network simulations for different scenarios

Introduction: Code Structure



Front-end

Front-end: User Interface

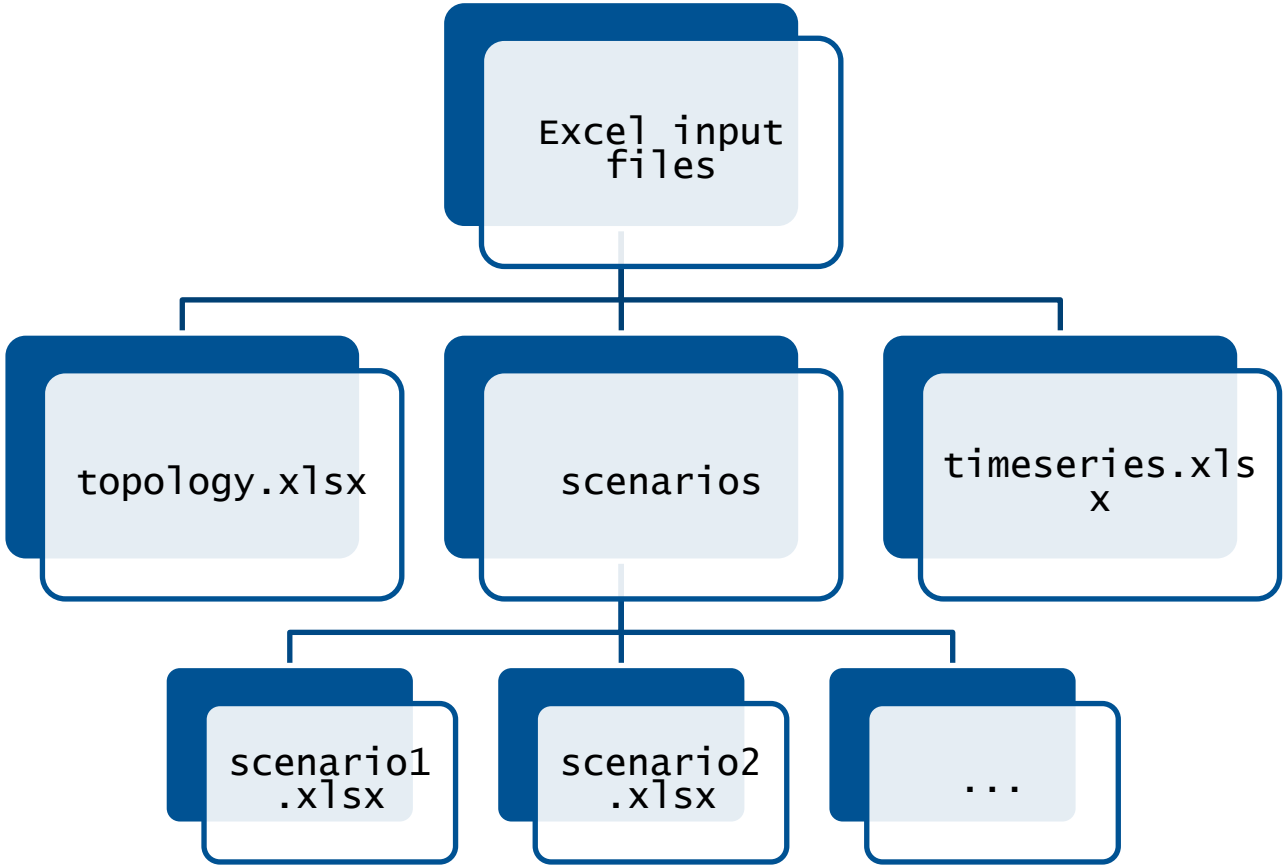
Excel input files

- Define network topology and scenarios
- If do time series analysis: time steps Excel file needed

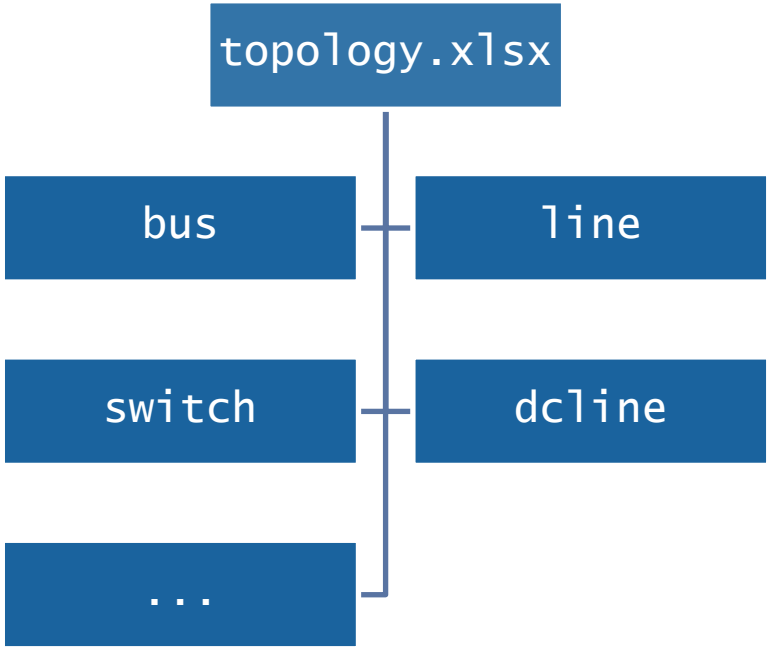
Python starting script

- Read the Excel input files and executes the toolbox

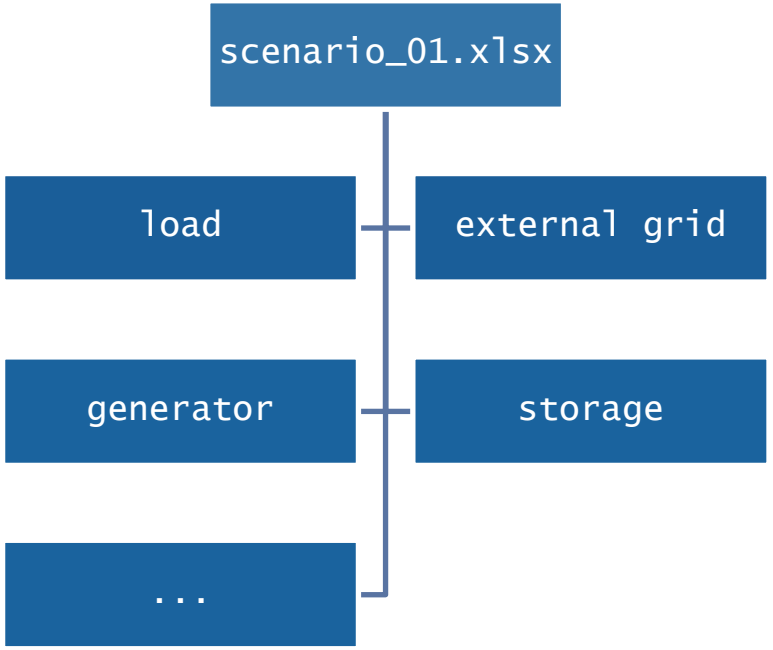
Front-end: Excel Input Files



Front-end: Excel Input Files

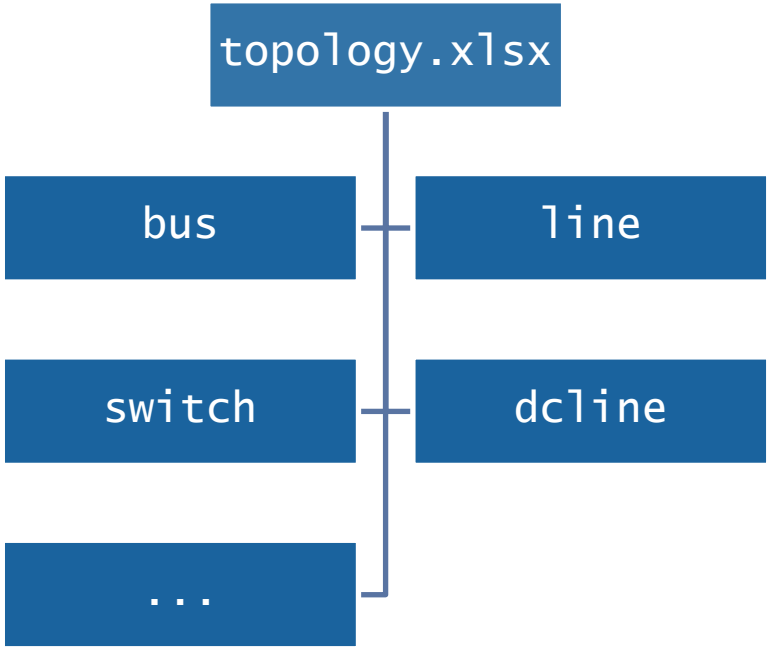


defines the graphical topology of the network

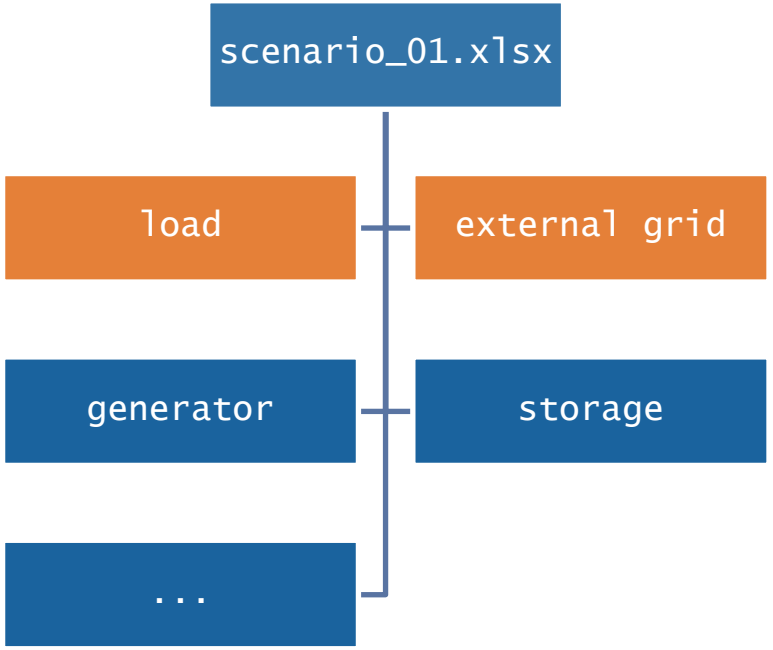


define the load/generation situation of the network

Front-end: Excel Input Files

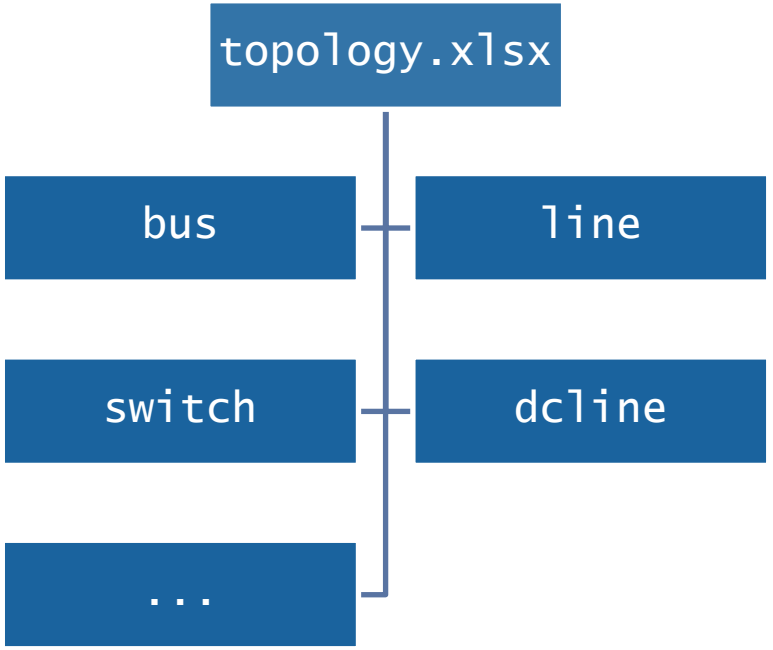


defines the graphical topology of the network

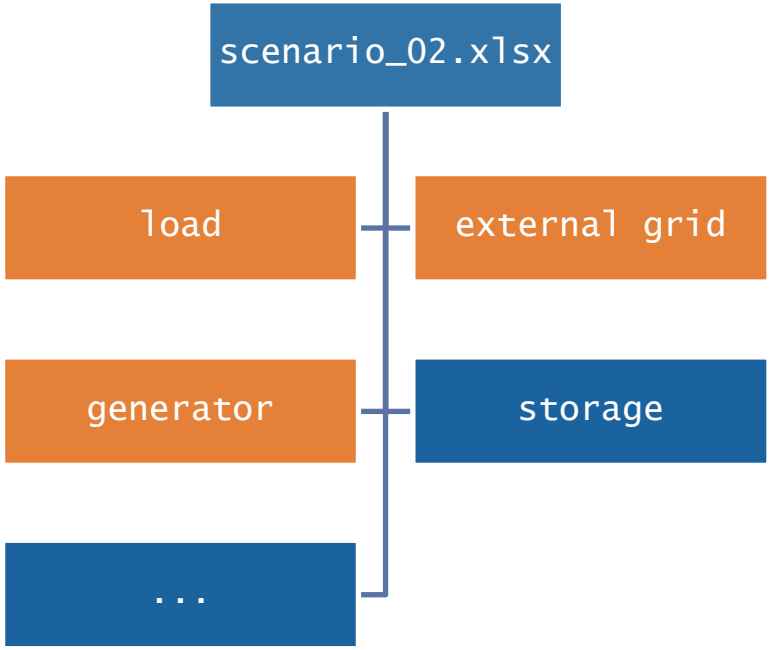


define the load/generation situation of the network

Front-end: Excel Input Files

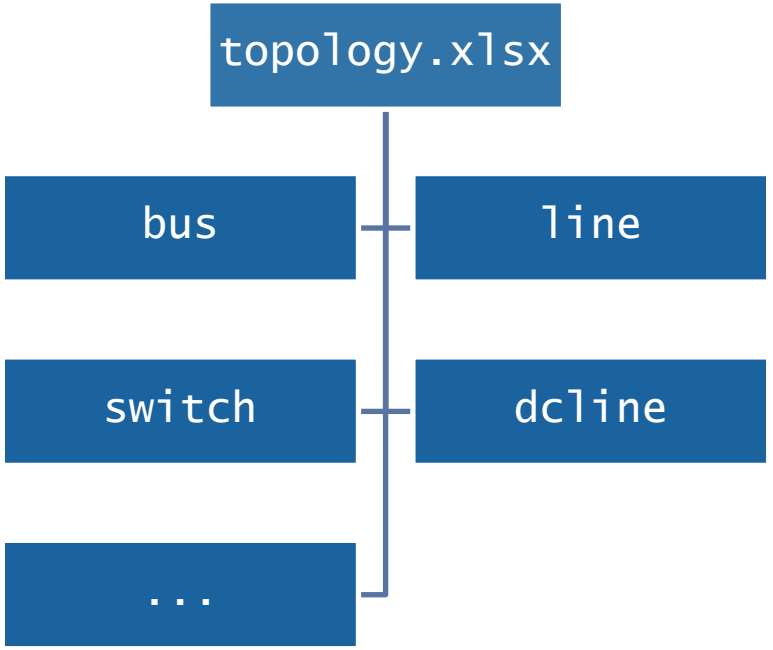


defines the graphical topology of the network

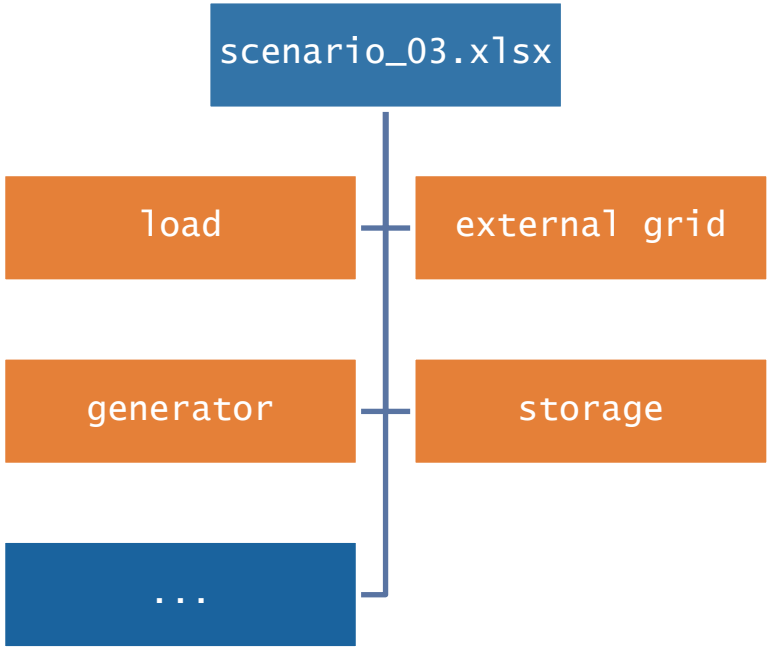


define the load/generation situation of the network

Front-end: Excel Input Files

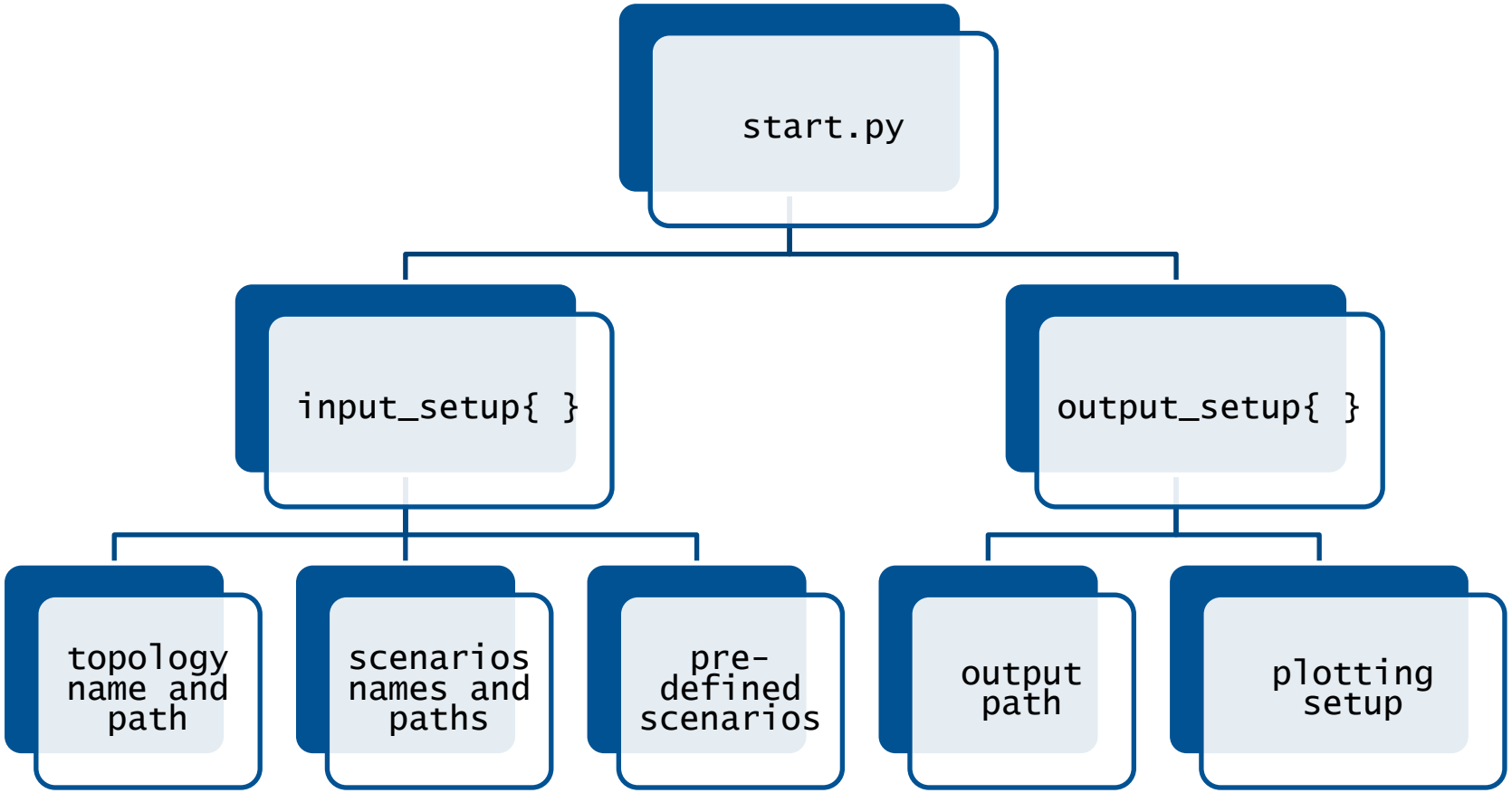


defines the graphical topology of the network



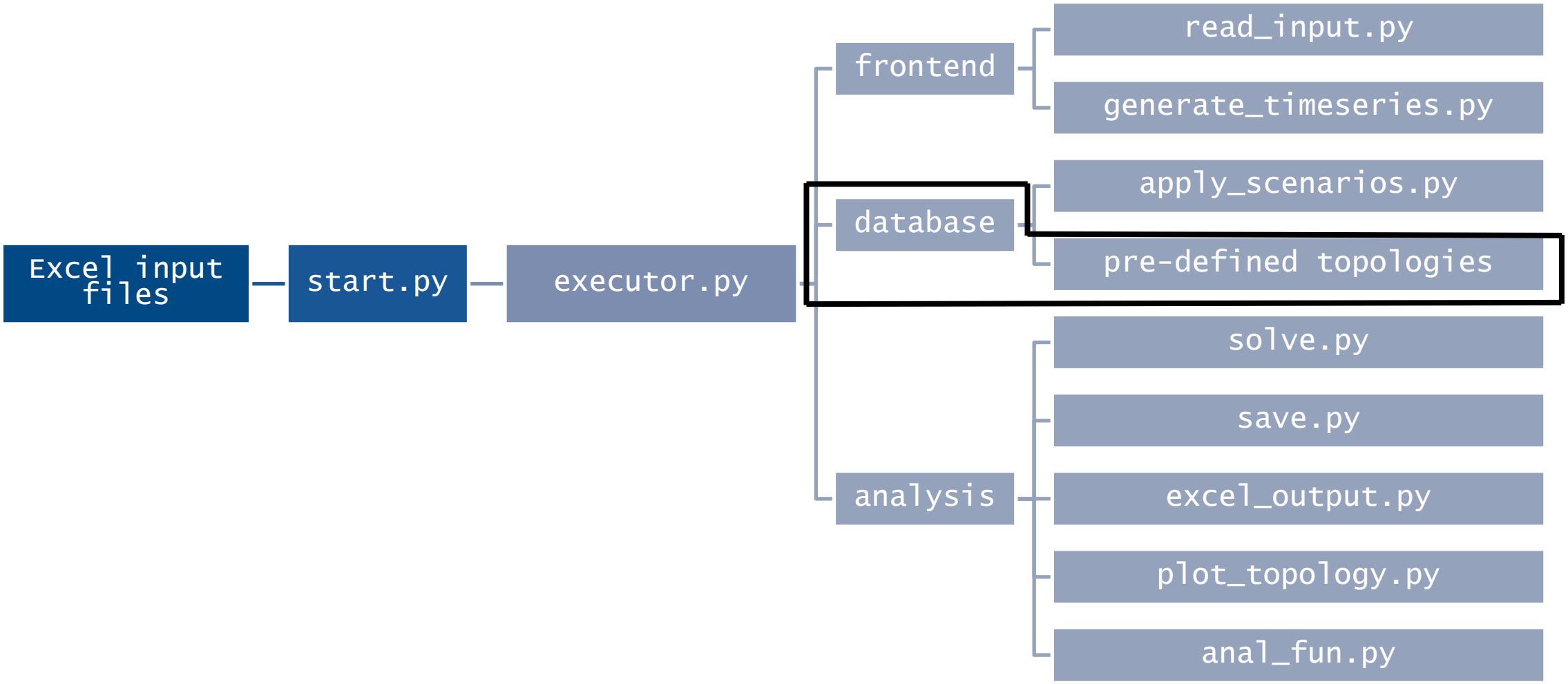
define the load/generation situation of the network

Front-end: Python Starting Script



Topology

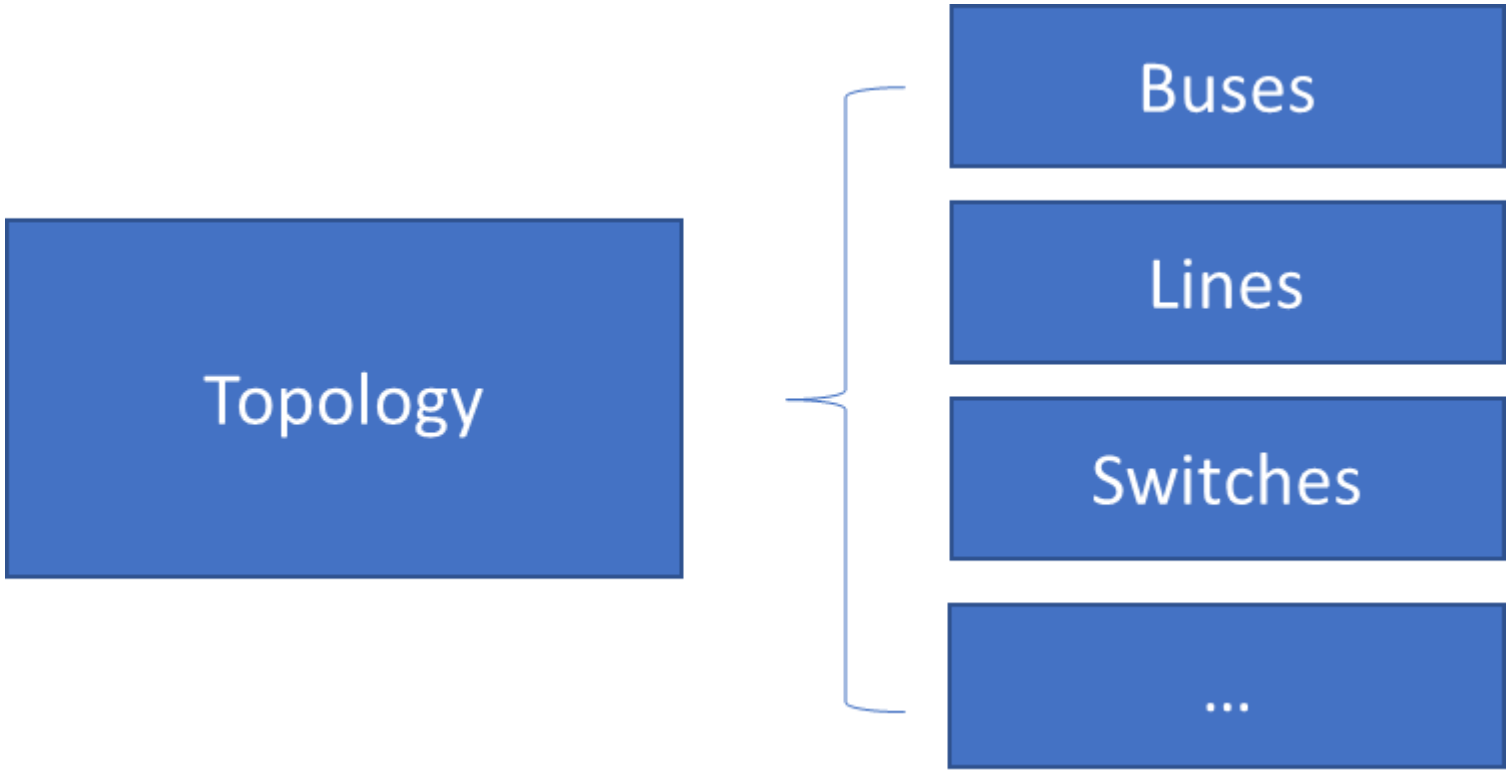
Topology: Code Structure



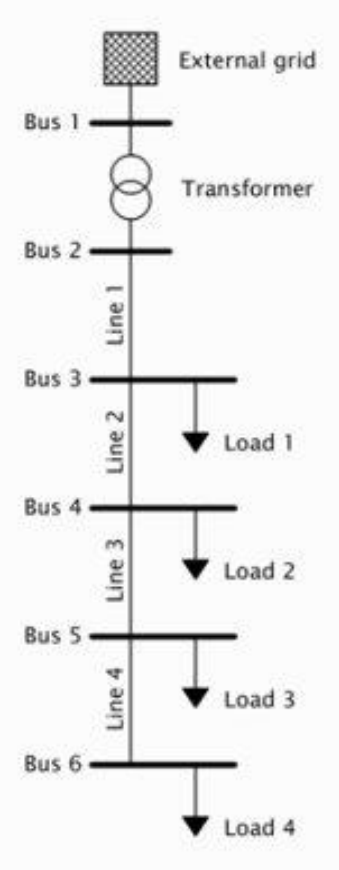
Topology: Objective

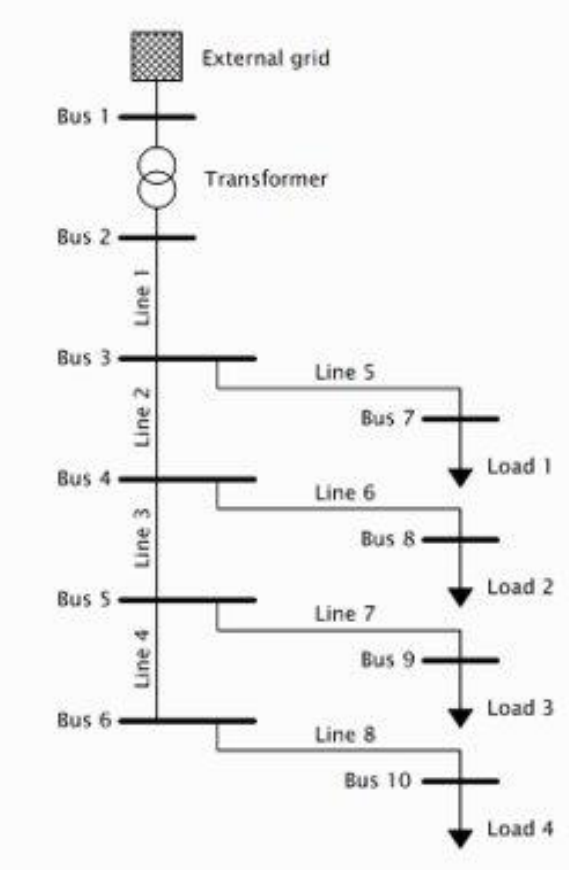
- Topologies are the different architectures that we can have for a Network. In our case we are considering electrical Networks.
- Exploring the topologies is to discover the main structure of the topologies, for example how many lines are there, or how many buses, or if there are some switches etc...
- Implement a variety of topologies

Topology: Main elements



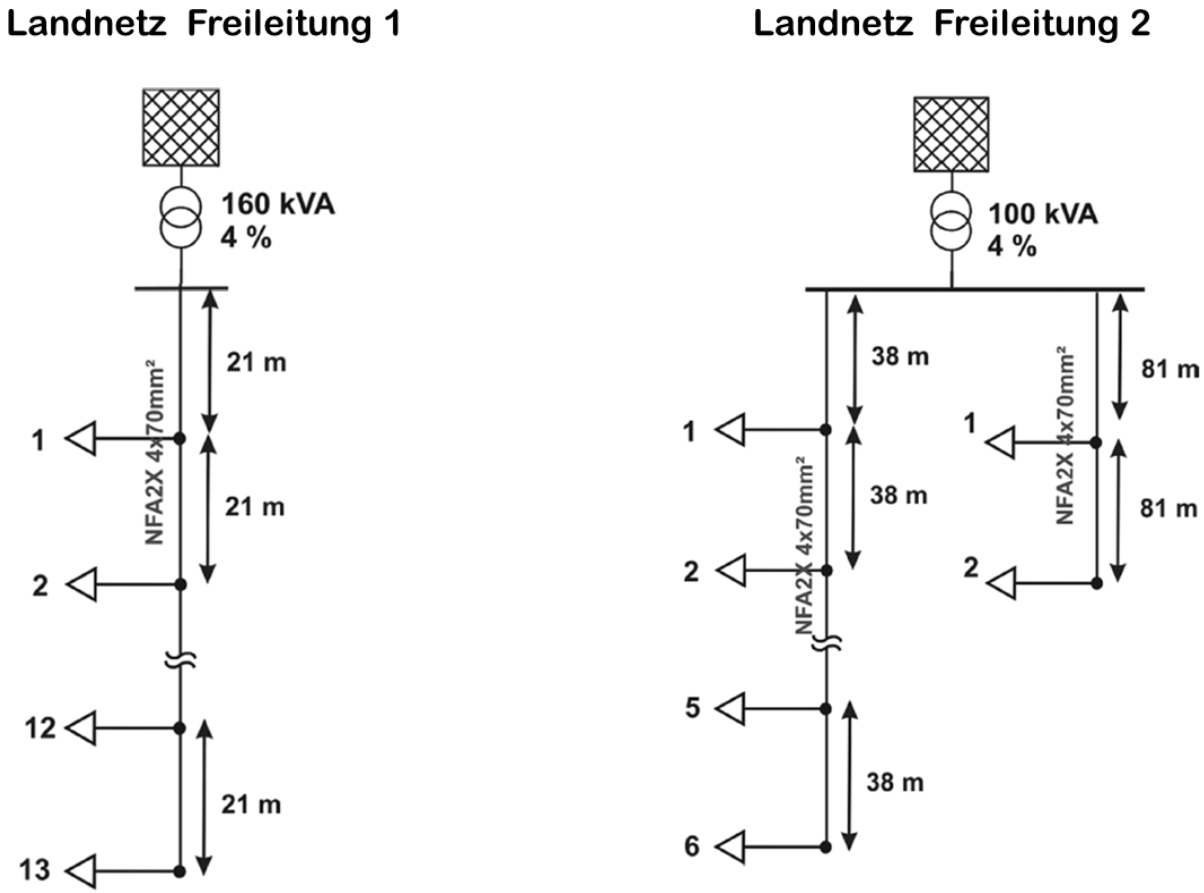
Topology: Four load branch vs Four loads with branches out





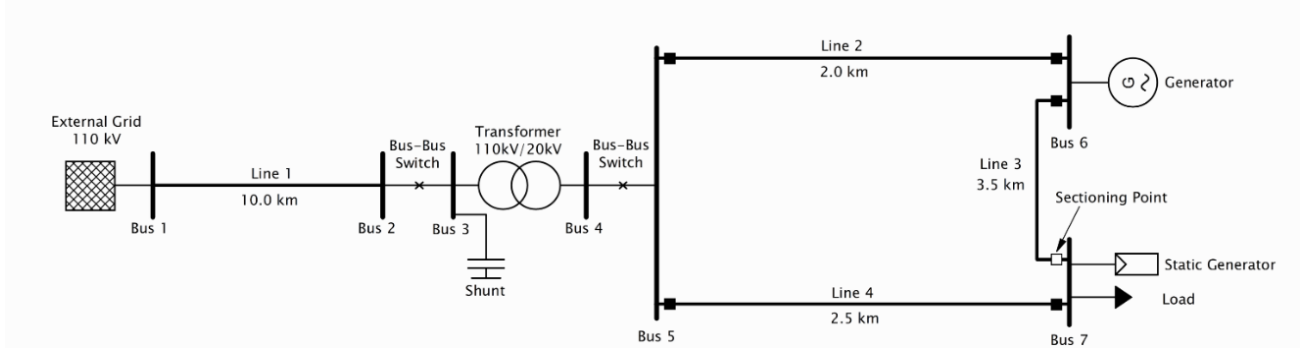
https://pandapower.readthedocs.io/en/v2.2.0/networks/test.html

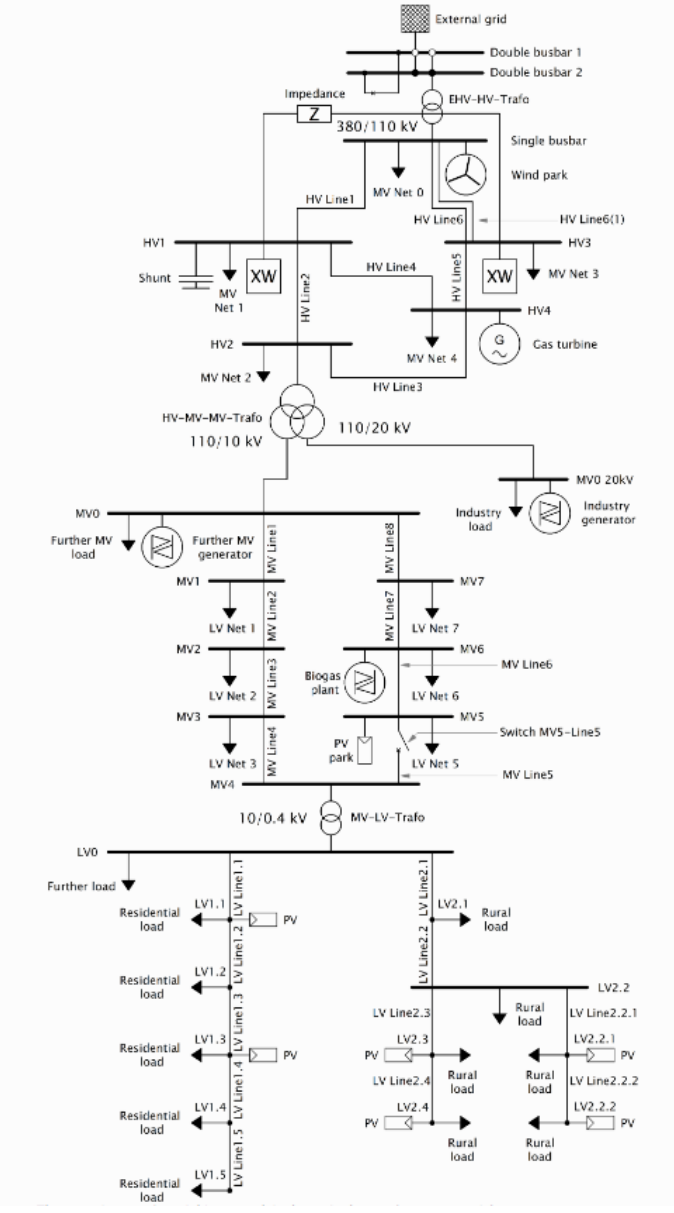
Topology: Kerber Landnetze



https://pandapower.readthedocs.io/en/v2.1.0/networks/kerber.html

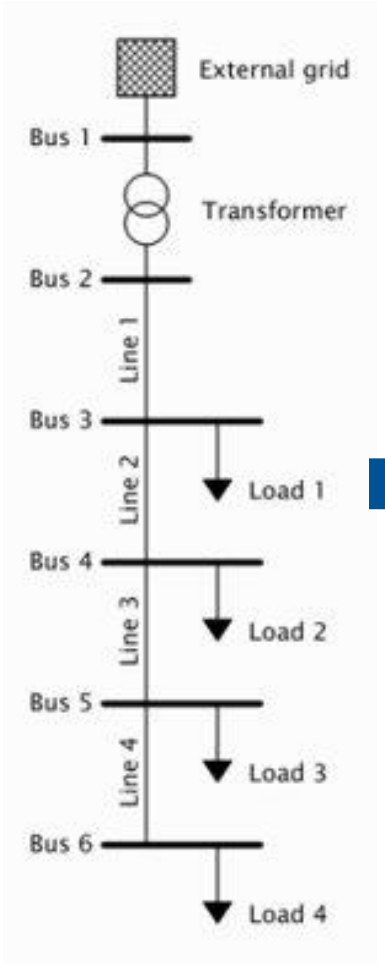
Topology: Simple example network vs Multi-Voltage Level Example Network

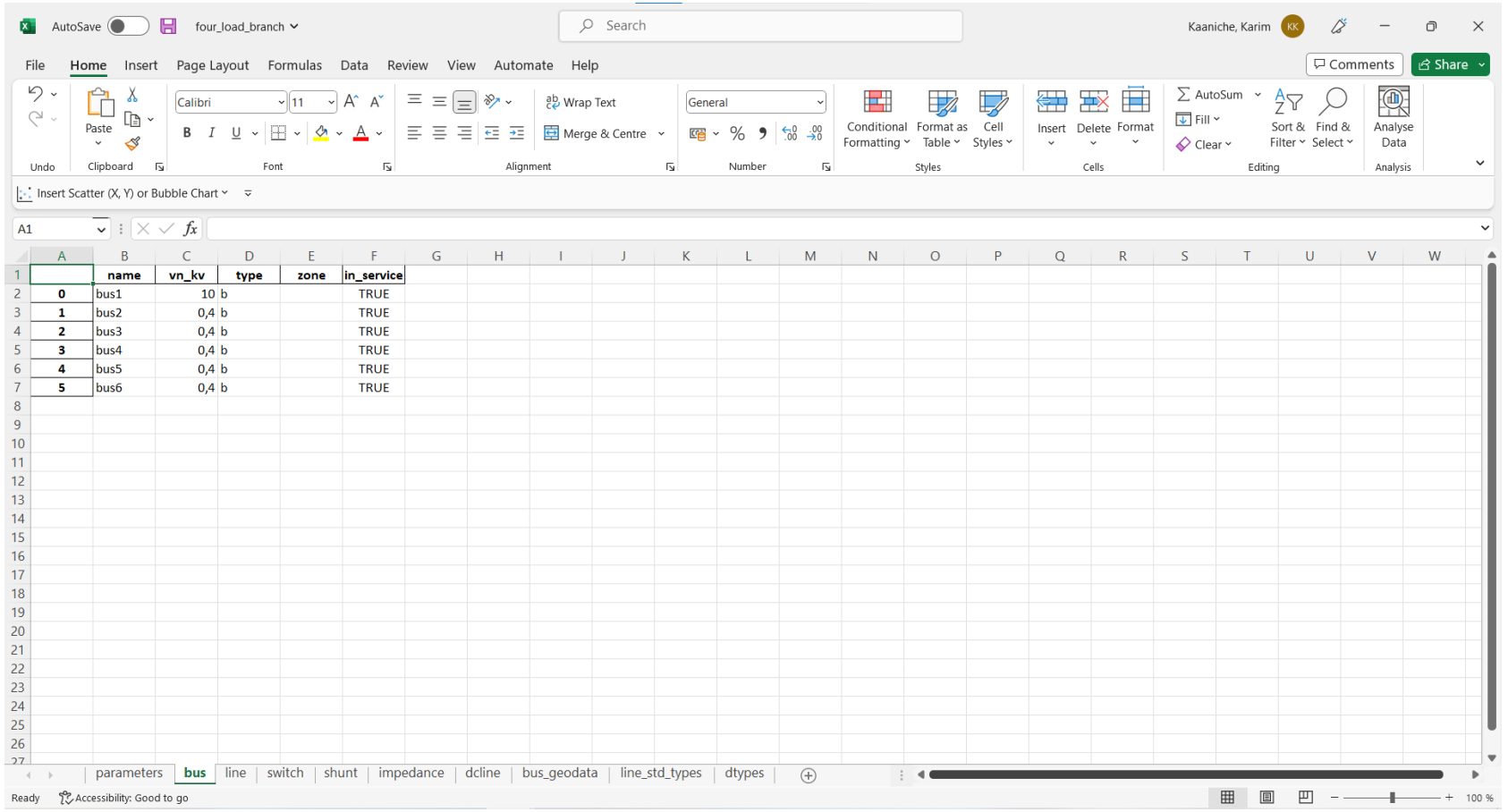




https://pandapower.readthedocs.io/en/latest/networks/example.html

Topology: From a network Topology to Excel





AutoSave | four_load_branch | Search | Kaaniche, Karim

File Home Insert Page Layout Formulas Data Review View Automate Help | Comments | Share

Insert Scatter (X, Y) or Bubble Chart

A1

	A	name	vn_kv	type	zone	in_service
2	0	bus1	10	b		TRUE
3	1	bus2	0,4	b		TRUE
4	2	bus3	0,4	b		TRUE
5	3	bus4	0,4	b		TRUE
6	4	bus5	0,4	b		TRUE
7	5	bus6	0,4	b		TRUE

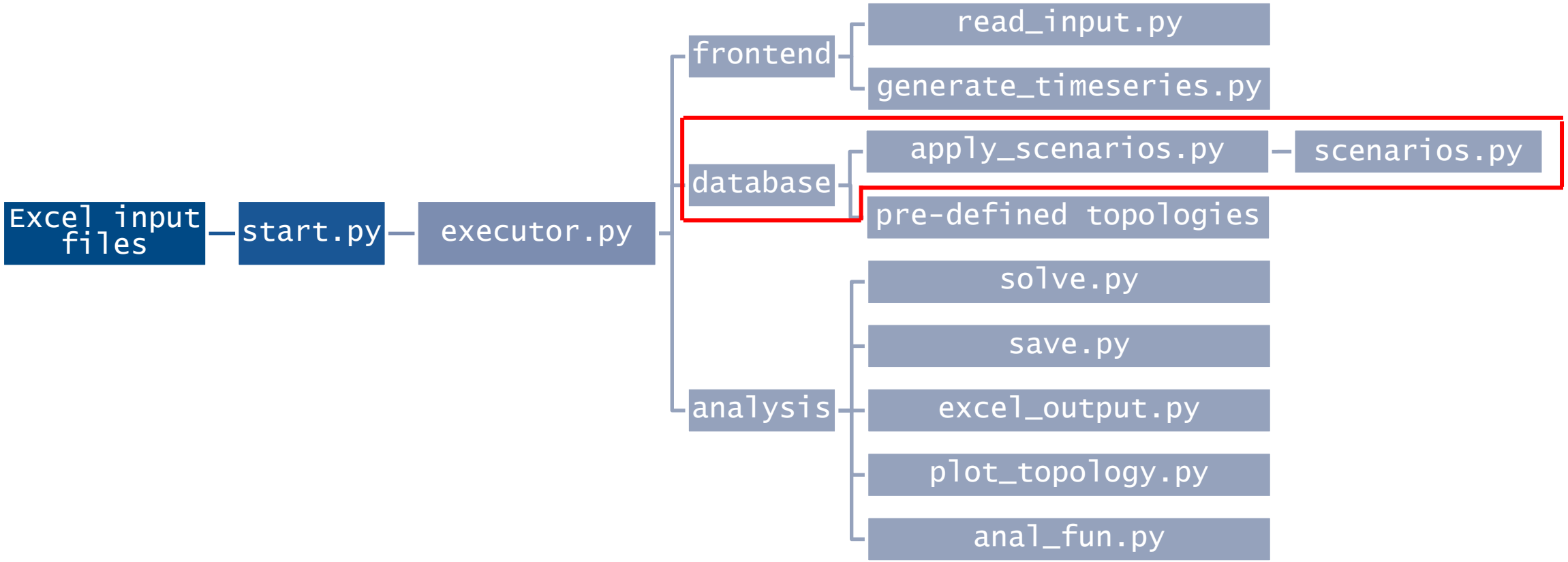
parameters | bus | line | switch | shunt | impedance | dcline | bus_geodata | line_std_types | dtypes

Ready | Accessibility: Good to go | 100 %

https://pandapower.readthedocs.io/en/v2.2.0/networks/test.html

Scenario

Scenario: Code Structure



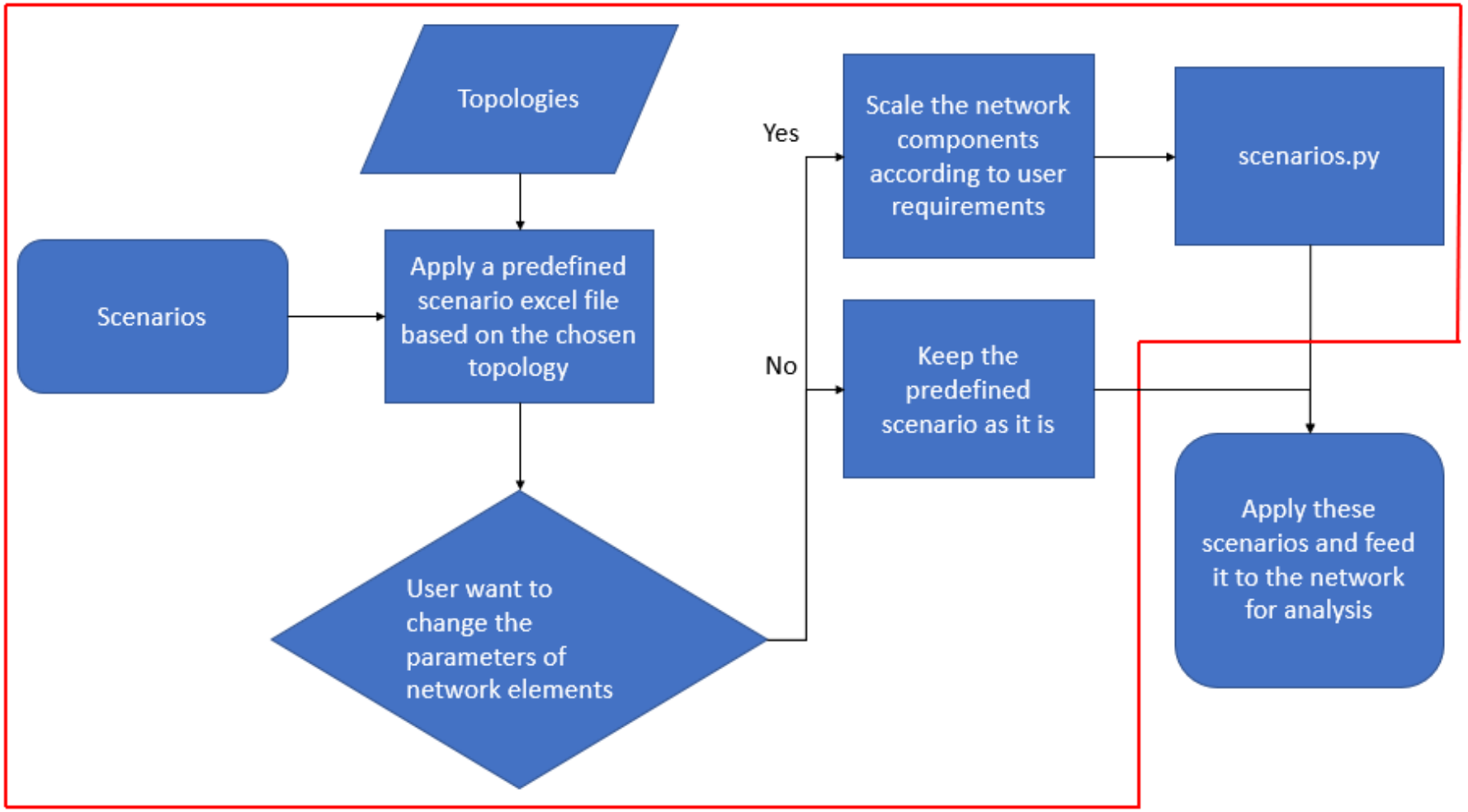
Scenario: Objective

- Modify the parameters of network elements that are flexible to alter in real life
- Providing the user with the freedom to experiment with these elements and how they can affect results in the power flow analysis
- Uses python-based functions
- Allows input from the user to modify (scale) the predefined scenarios

Scenario: Network Elements

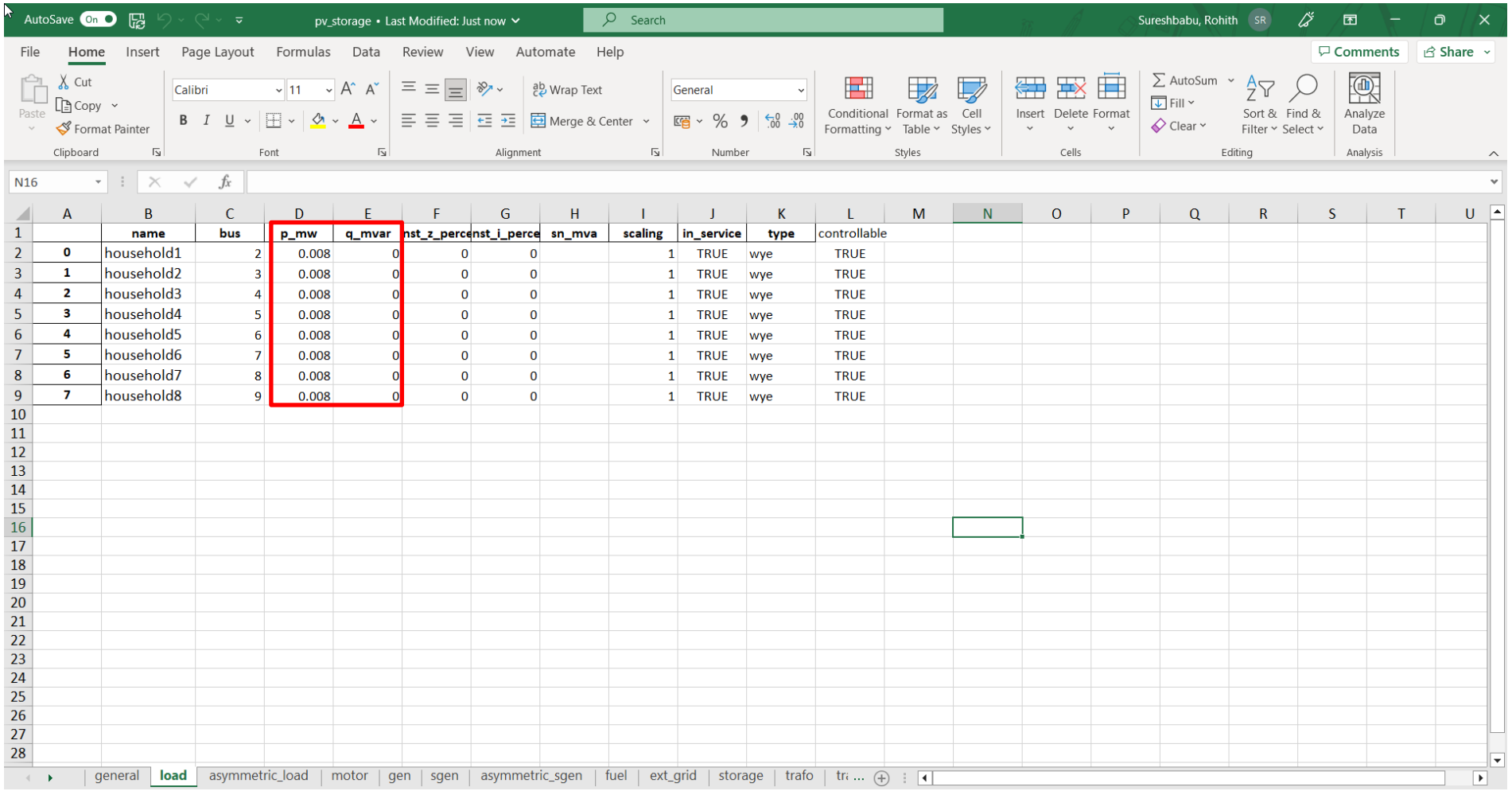
Elements	Type	Parameter to be altered	Scaling Values (%)
Static Generator	Photovoltaic Wind Conventional Power Plants	Active and Reactive power	0 - 200
Load	-	Active and Reactive power	0 - 200
Transformer	-	Apparent power	0 - 200
Lines	-	Maximum thermal current Parallel lines	0 - 195 >=2
Storage	-	Maximum storable energy	0 - 100

Scenario: Logic flow



Scenario: Scaling the Parameters

Example: Load



	name	bus	p_mw	q_mvar	nst_z_perce	nst_i_perce	sn_mva	scaling	in_service	type	controllable
0	household1	2	0.008	0	0	0		1	TRUE	wye	TRUE
1	household2	3	0.008	0	0	0		1	TRUE	wye	TRUE
2	household3	4	0.008	0	0	0		1	TRUE	wye	TRUE
3	household4	5	0.008	0	0	0		1	TRUE	wye	TRUE
4	household5	6	0.008	0	0	0		1	TRUE	wye	TRUE
5	household6	7	0.008	0	0	0		1	TRUE	wye	TRUE
6	household7	8	0.008	0	0	0		1	TRUE	wye	TRUE
7	household8	9	0.008	0	0	0		1	TRUE	wye	TRUE

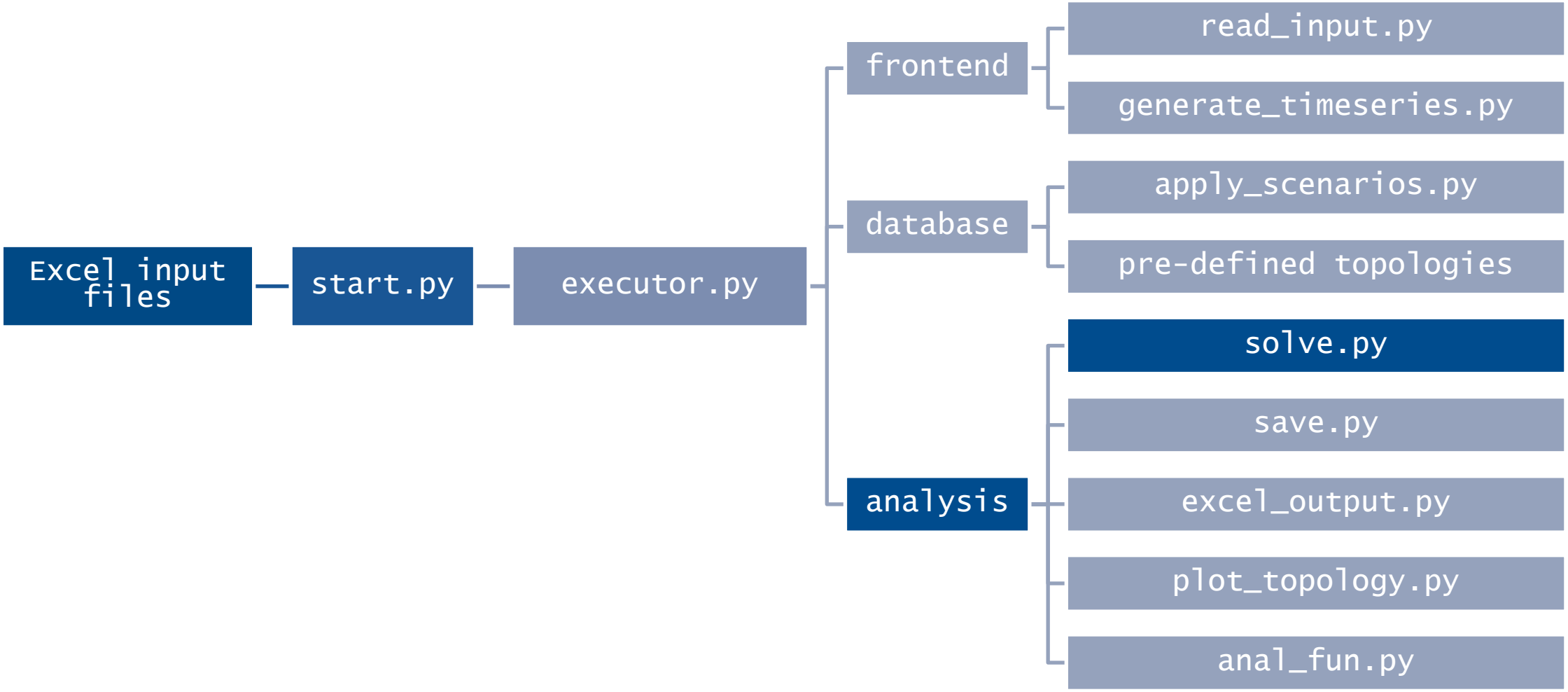
p_mw	q_mvar
0.008	0
0.008	0
0.008	0
0.008	0
0.008	0
0.008	0
0.008	0
0.008	0

User Input of 0.5

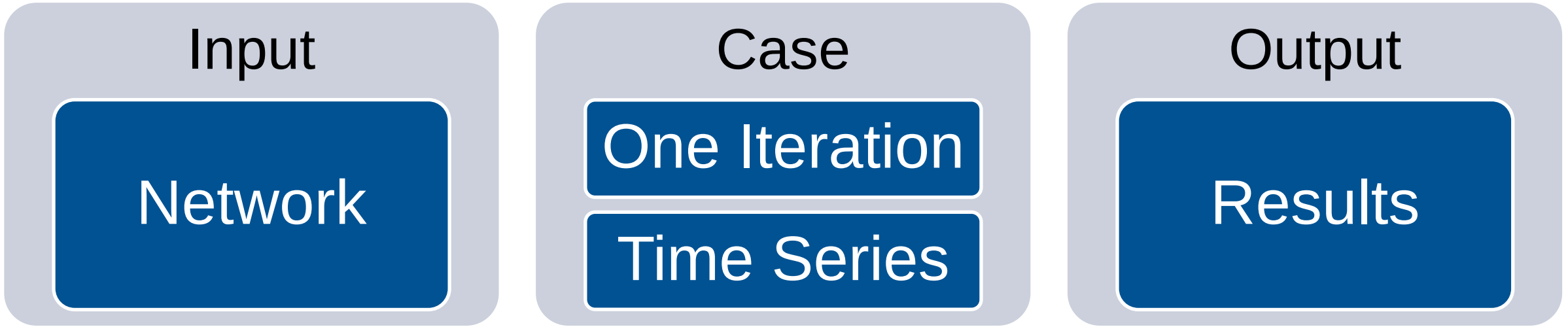
p_mw	q_mvar
0.004	0
0.004	0
0.004	0
0.004	0
0.004	0
0.004	0
0.004	0
0.004	0

Analysis

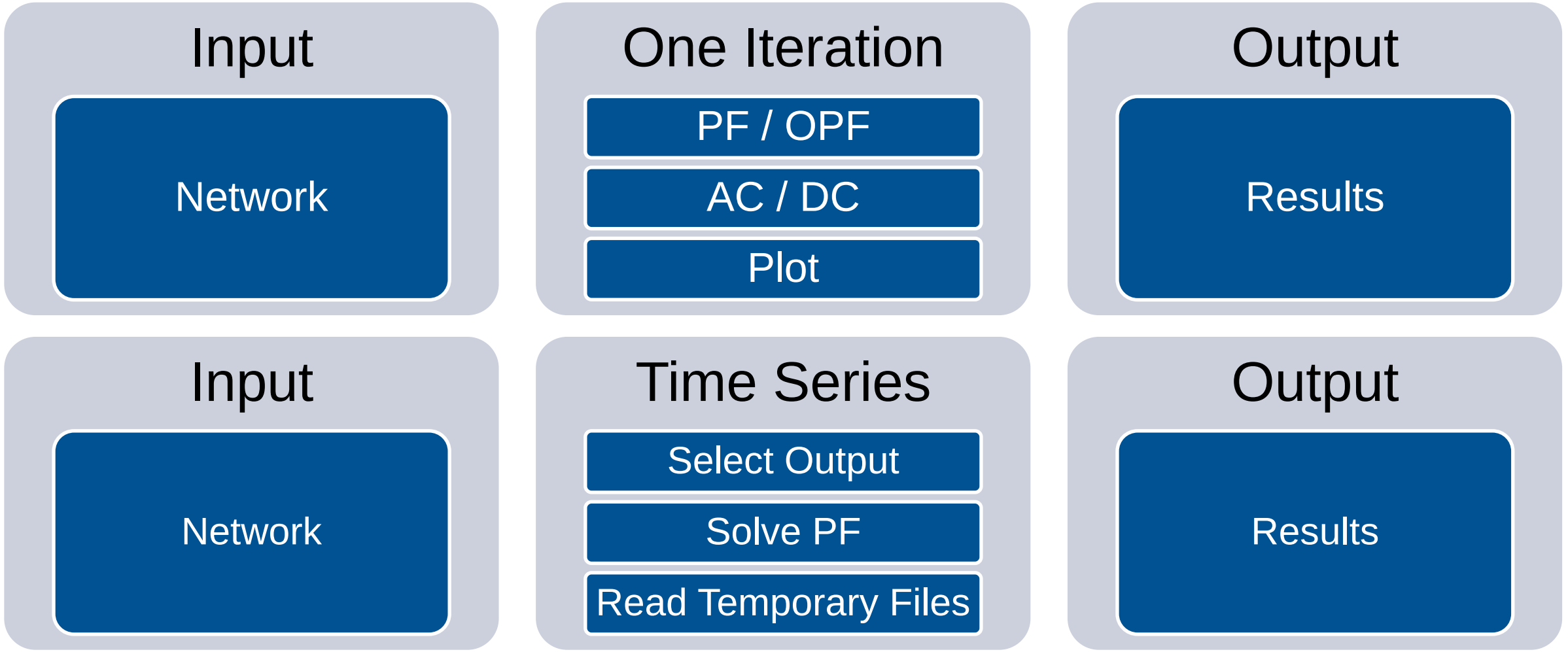
Analysis: Overview



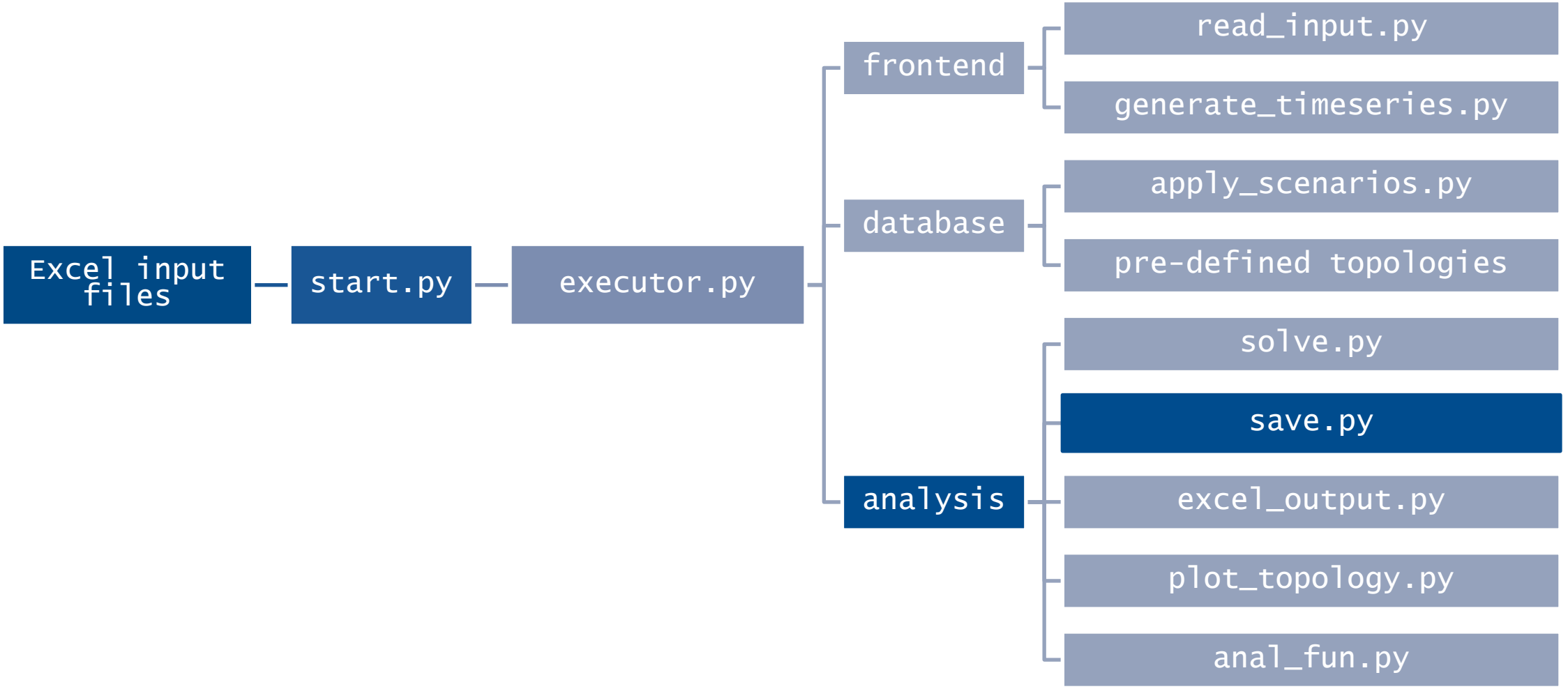
Analysis: solve.py



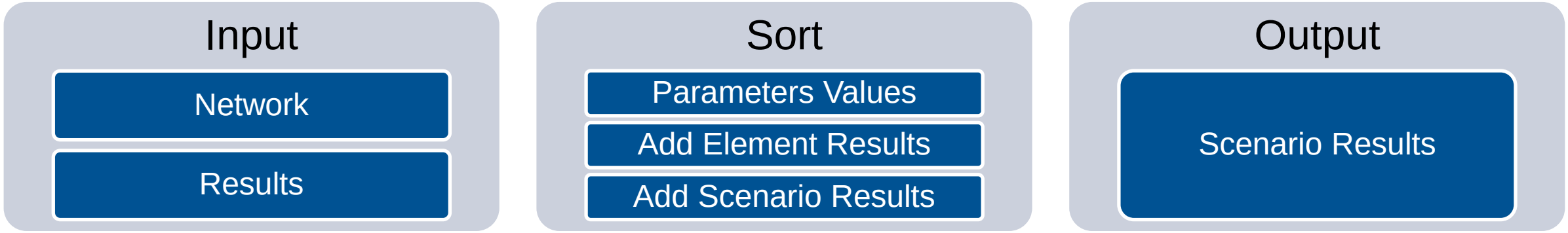
Analysis: solve.py

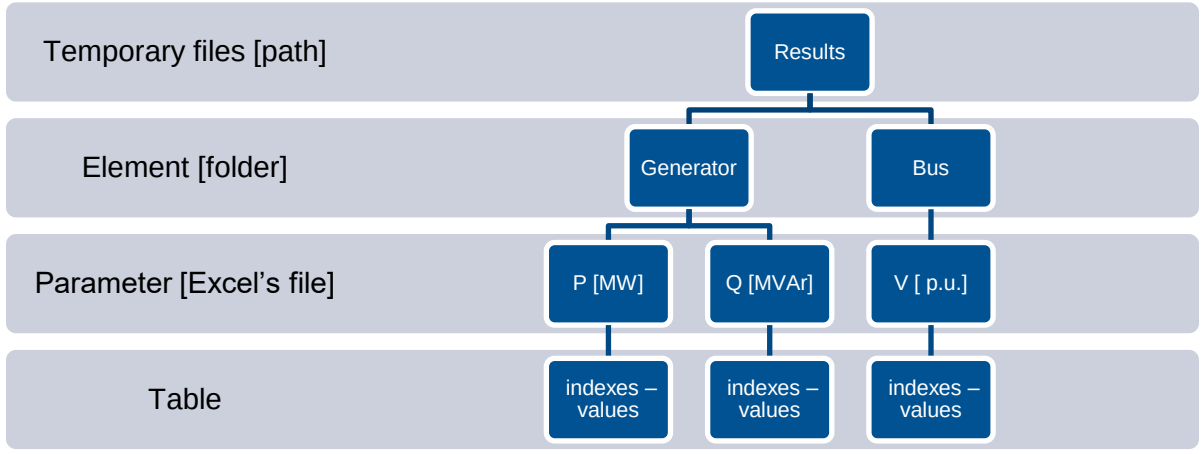


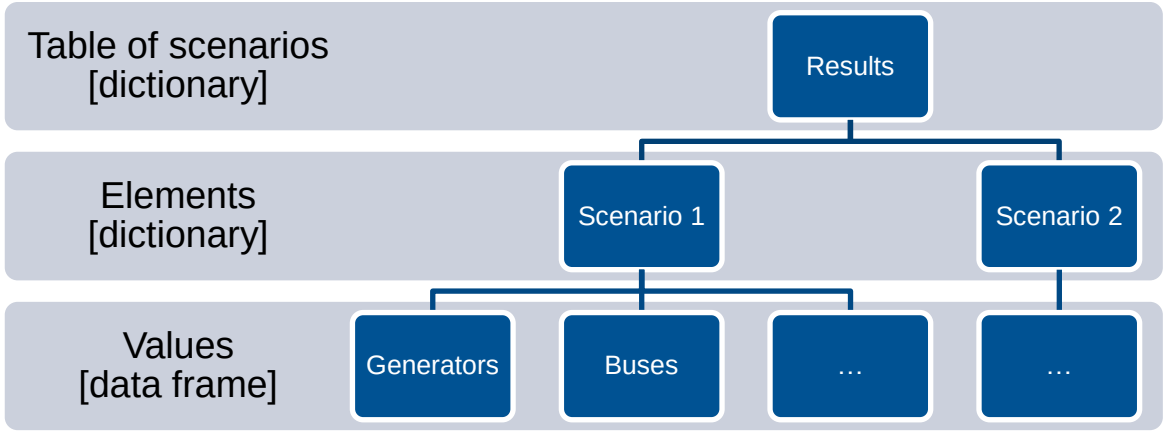
Analysis: Overview



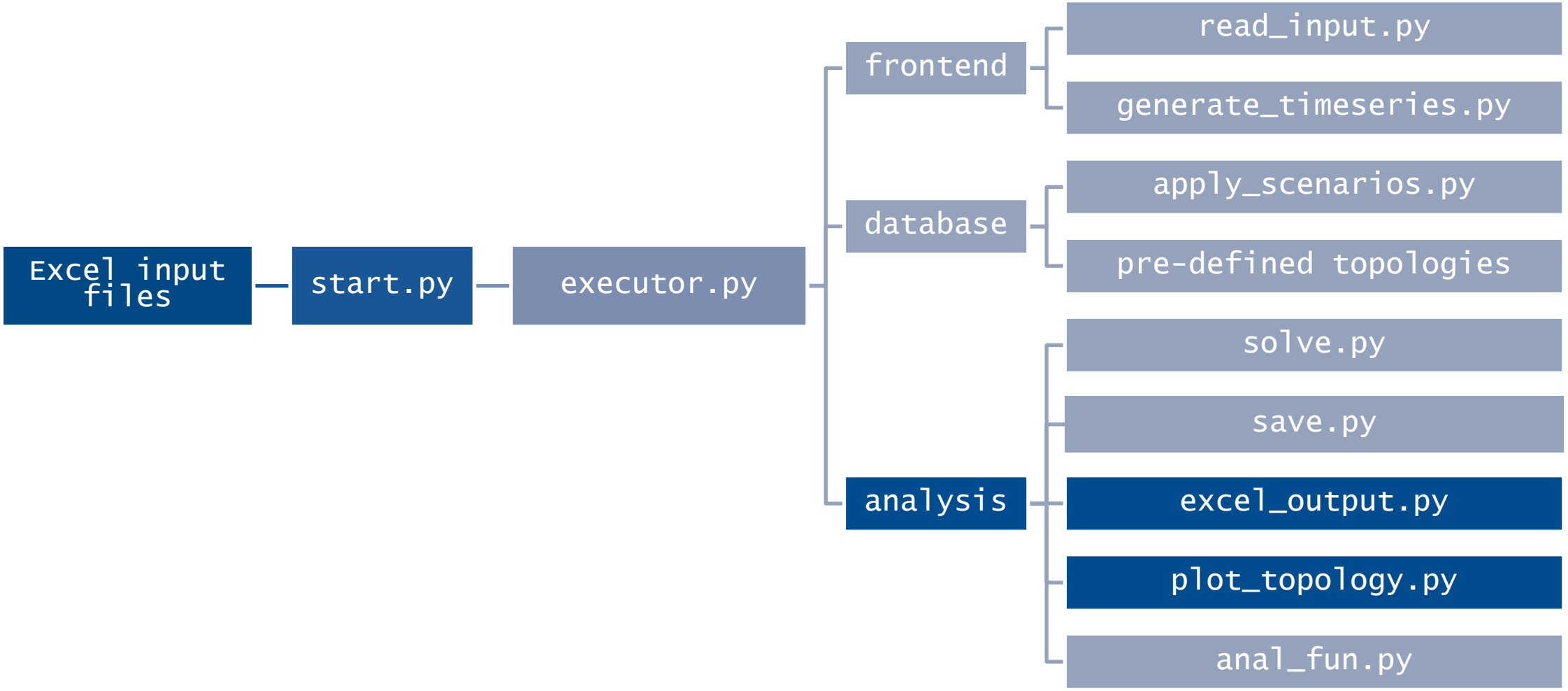
Analysis: save.py



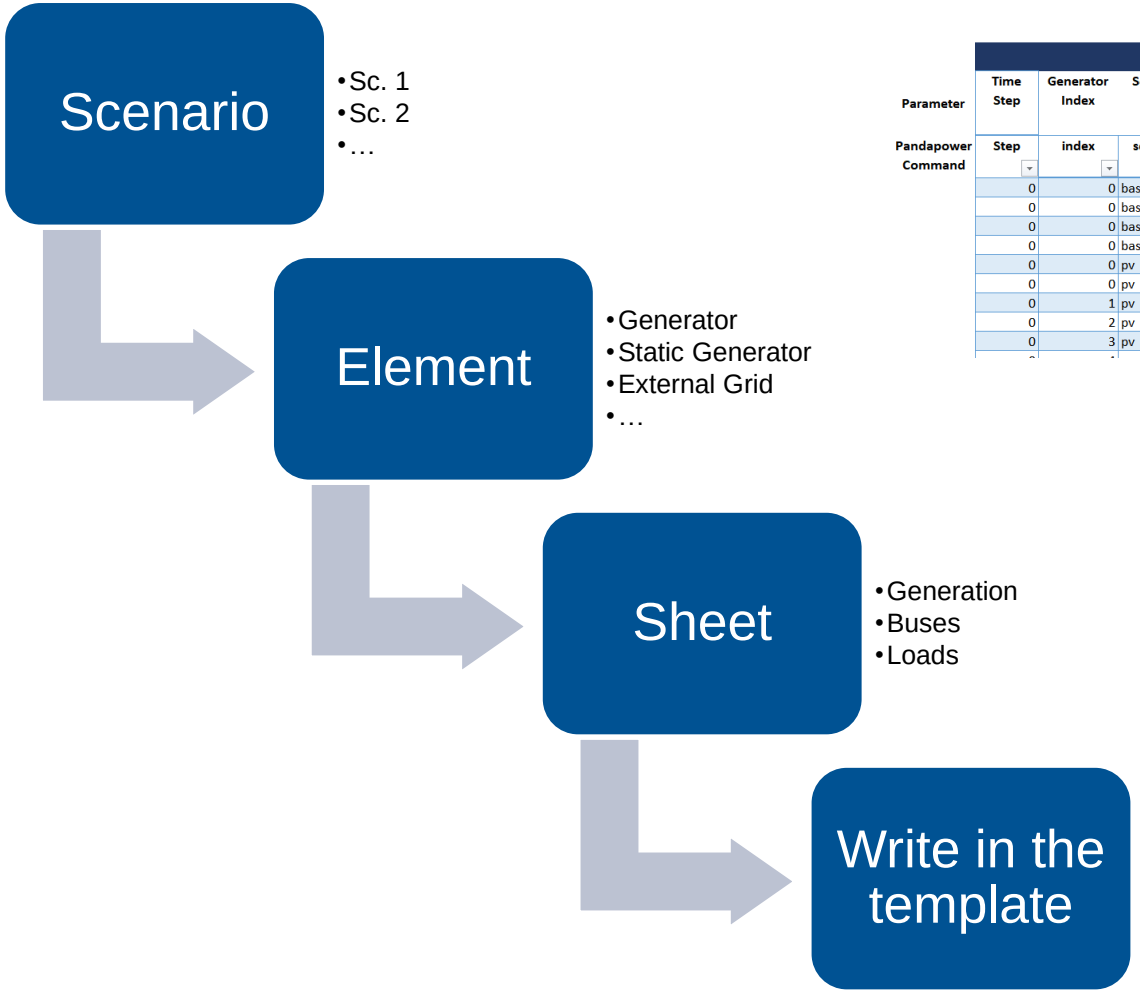




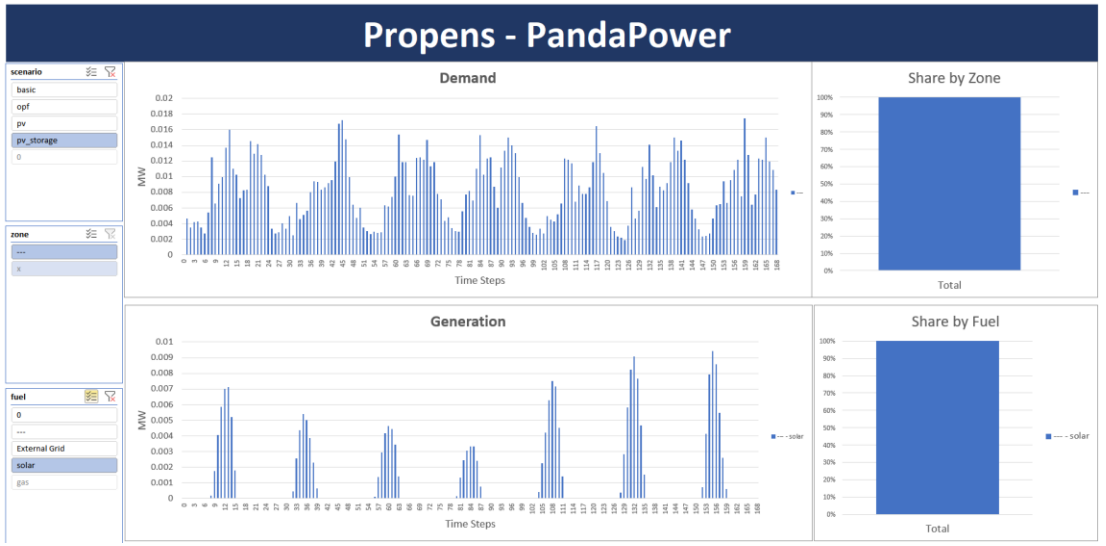
Analysis: Overview



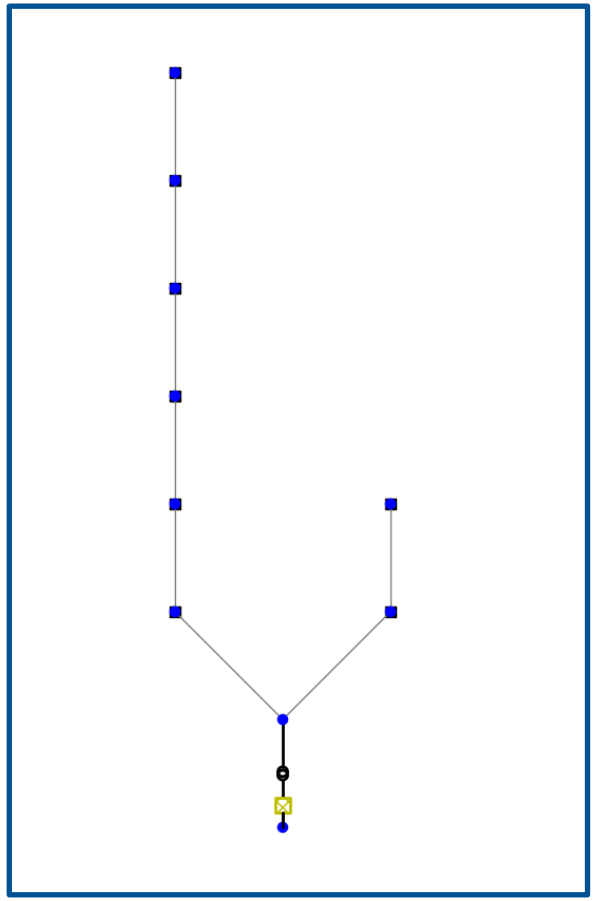
Analysis: excel_output.py



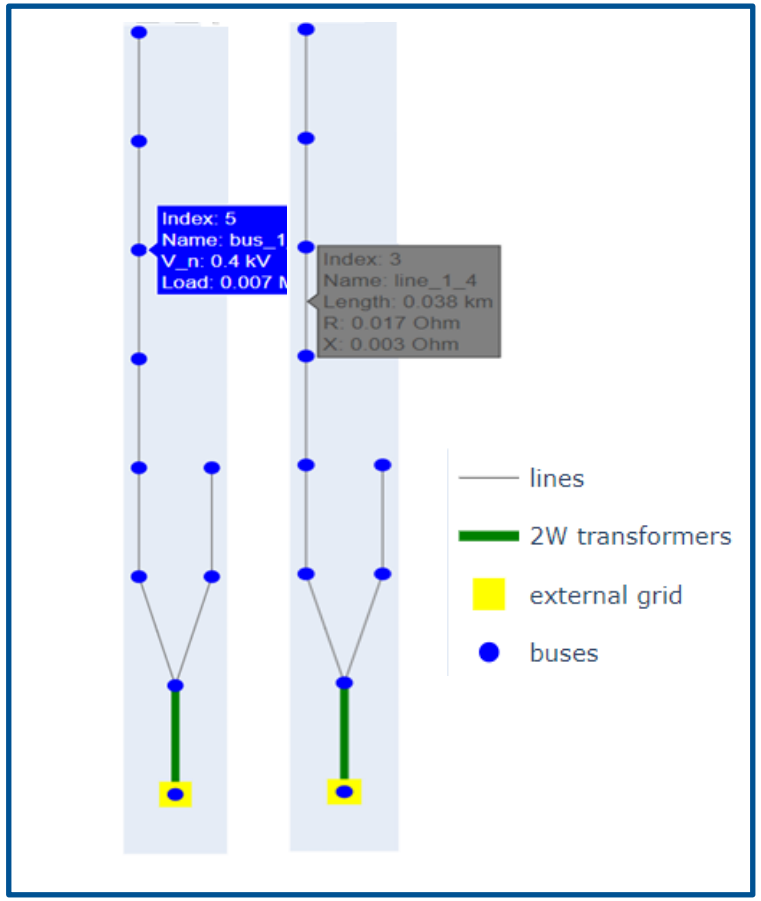
Parameter	Generators															
	Time Step	Generator Index	Scenario Name	Zone	Name	Type	Bus Index	In Service	Fuel / Source	Voltage Setting Point [PV bus]	Maximum Active Power [MW]	Maximum Reactive Power	Minimum Active Power [MW]	Minimum Reactive Power [MVAr]	Active Power [MW]	Reactive Power [MVAr]
Pandapower Command	Step	index	scenario	zone	name	type	bus	in_service	fuel	vm_pu	max_p_mw	max_q_mw	min_p_mw	min_q_mvar	p_mw	q_mvar
	0	0	basic	0	0	gen	0	0	0	0	0	0	0	0	0	0
	0	0	basic	0	0	sgen	0	0	0	0	0	0	0	0	0	0
	0	0	basic	---	---	ext_grid	0	1	---	1	---	---	---	---	0.059038815	0.001410555
	0	0	basic	0	0	storage	0	0	0	0	0	0	0	0	0	0
	0	0	pv	0	0	gen	0	0	0	0	0	0	0	0	0	0
	0	0	pv	---	pv1	sgen	2	1	solar	---	---	---	---	---	0.009	0
	0	1	pv	---	pv2	sgen	3	1	solar	---	---	---	---	---	0.009	0
	0	2	pv	---	pv3	sgen	4	1	solar	---	---	---	---	---	0.009	0
	0	3	pv	---	pv4	sgen	5	1	solar	---	---	---	---	---	0.009	0



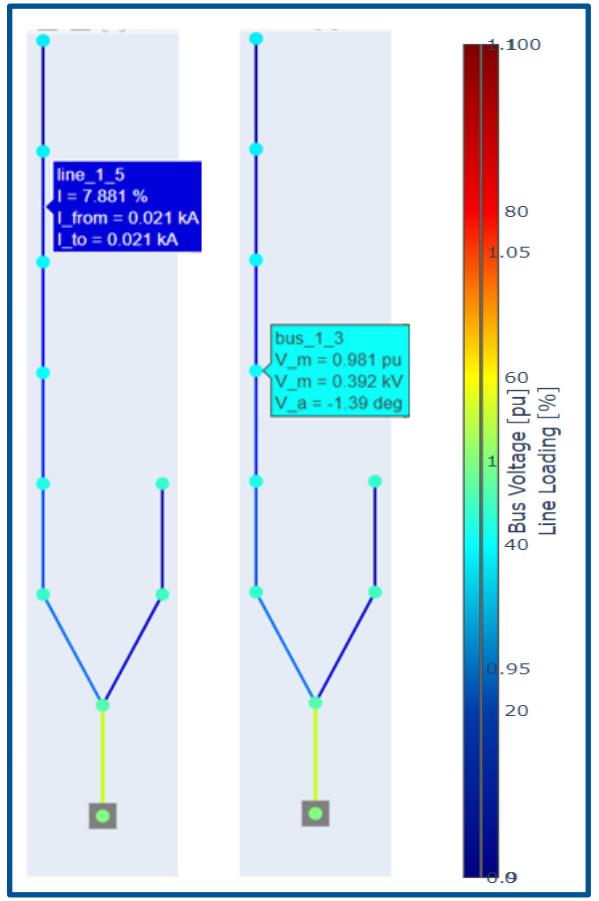
Analysis: simple and interactive plots



Simple Plot

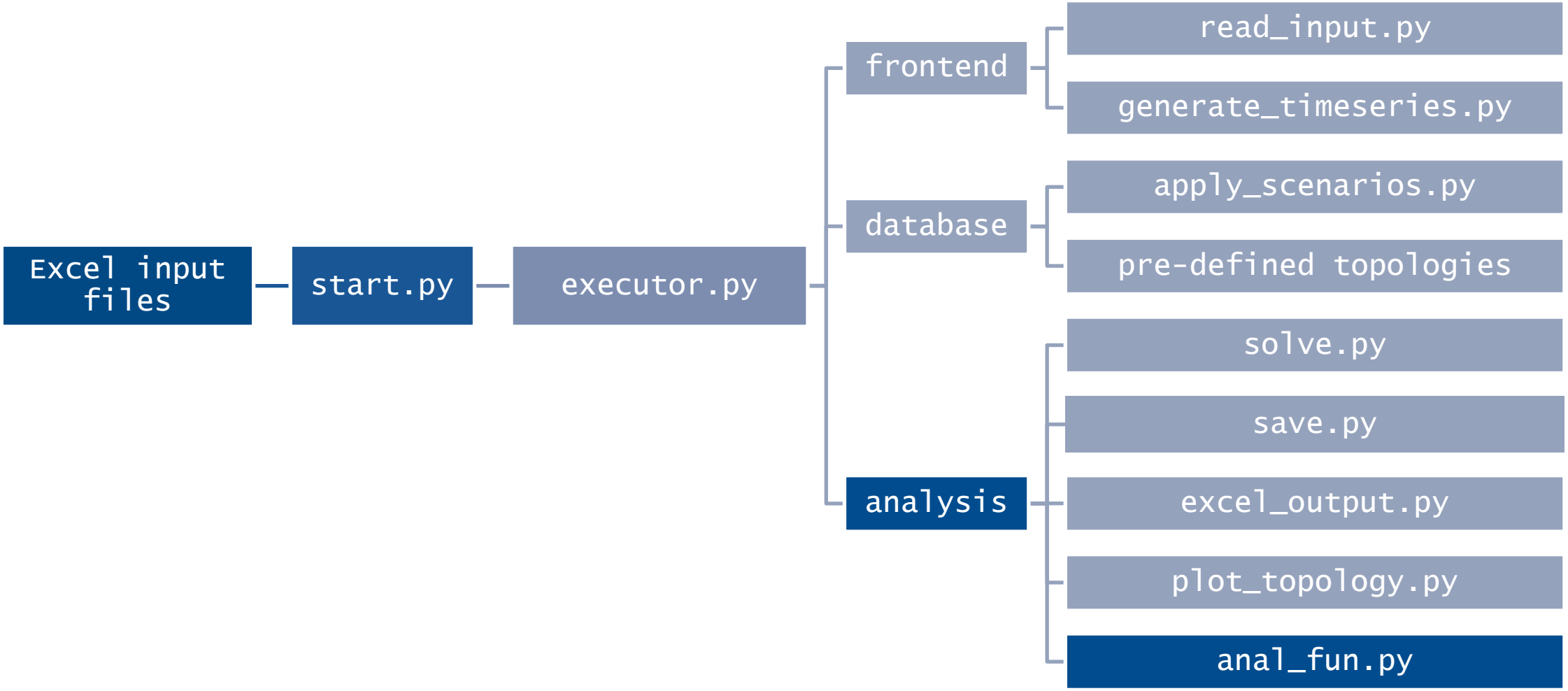


Interactive Plot



Interactive Heat Map Plot

Analysis: Overview



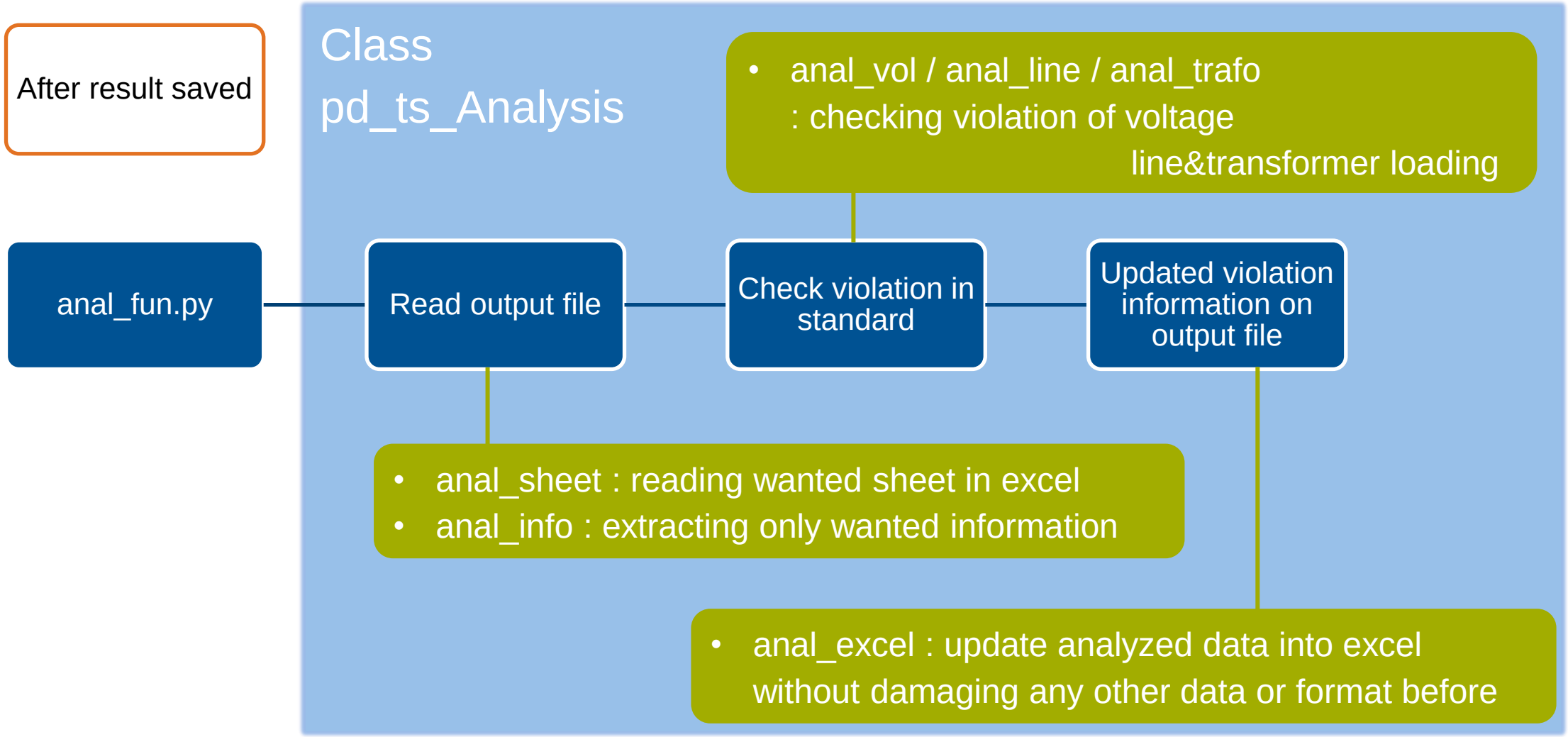
Analysis: Analysis Summary Objective

- Giving an overview of violation components to the user
- Provide the information of Time Step, Location of violation and Value of the violating component

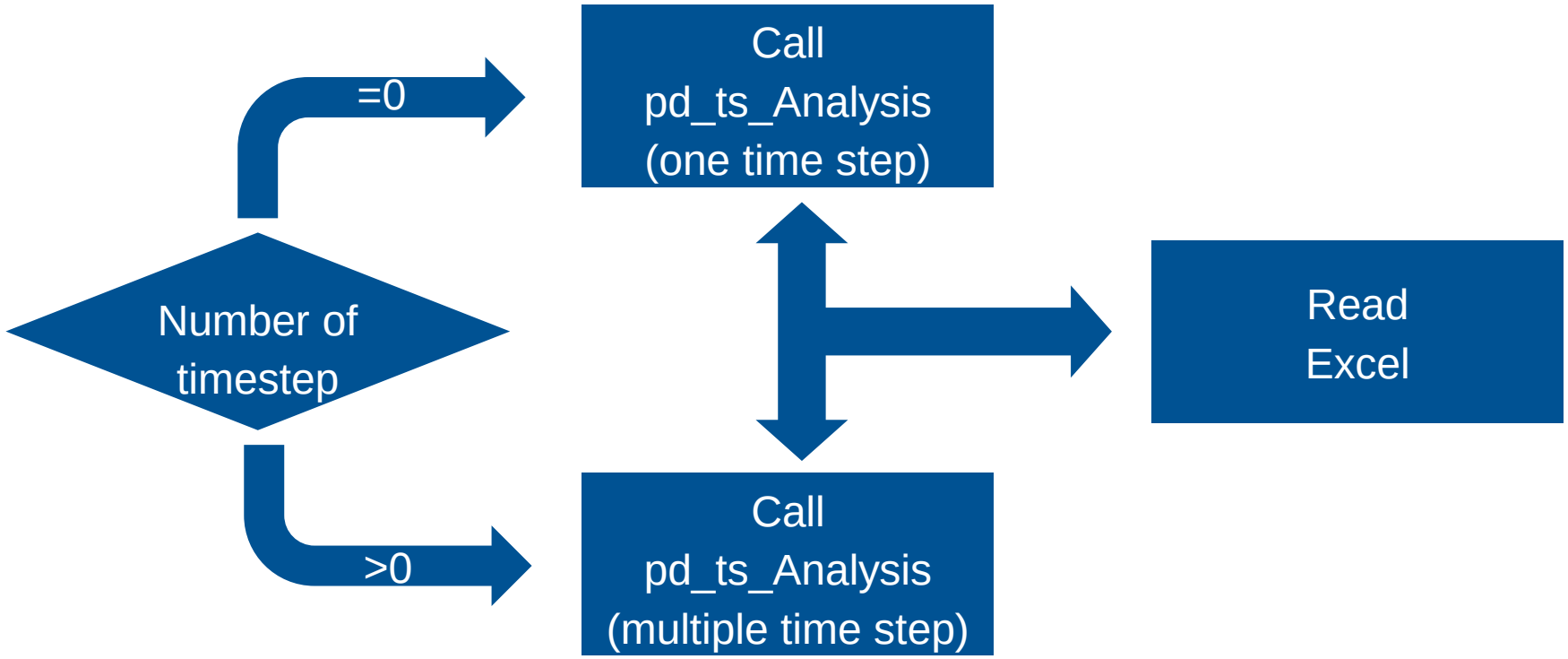
---

- Building a Function that is independent from other Analysis part
- Try using class feature for easy update
- Updating the information on existing result file without conflicting already existing features

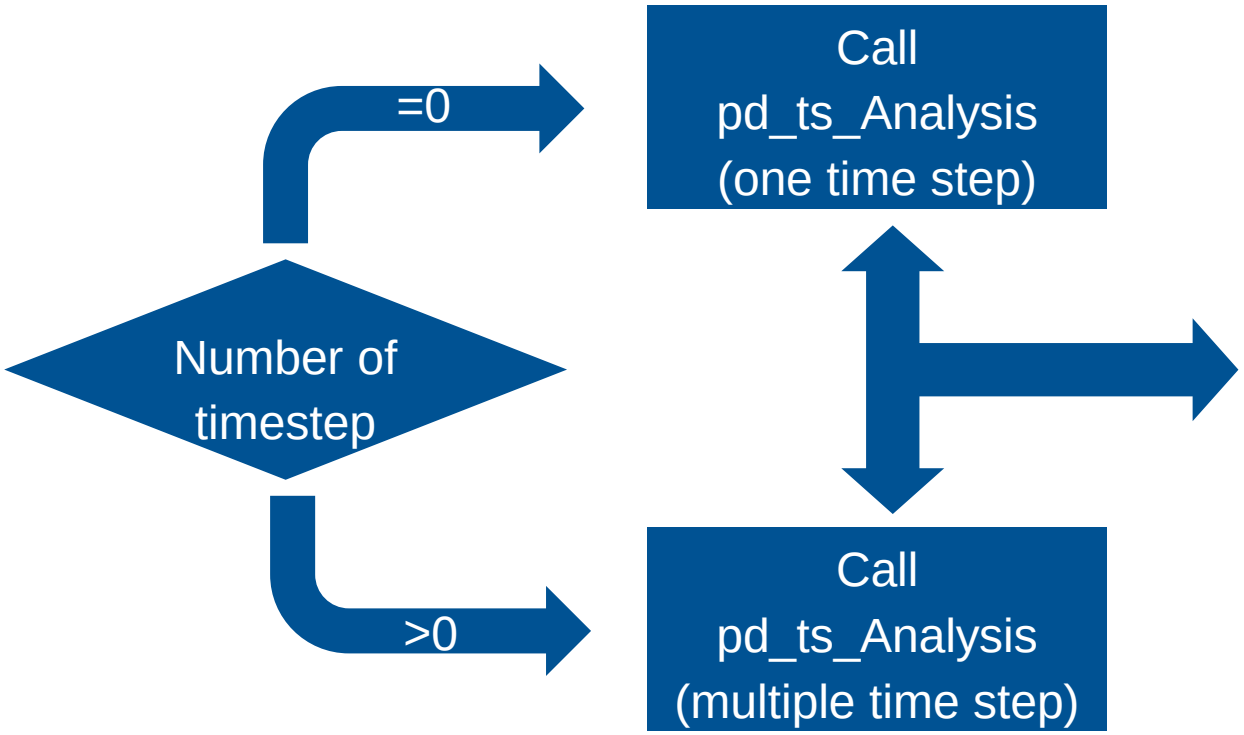
Analysis: Analysis Summary WorkFlow



Analysis: Time Series Flexibility



Analysis: Reading Excel(def anal_sheet)



Time Step	Bus Index	Scenario Name	Zone	Name	Voltage Level [kV]	In Service	Voltage [p.u]
step	index	scenario	zone	name	vn_kv	in_service	vm_pu
0	0	basic	---	Trafostatio	10	1	1
0	1	basic	---	main_busb	0.4	1	0.99265488
0	2	basic	---	bus_1_1	0.4	1	0.98802016
0	3	basic	---	bus_1_2	0.4	1	0.9841526
0	4	basic	---	bus_1_3	0.4	1	0.98105507
0	5	basic	---	bus_1_4	0.4	1	0.97872987
0	6	basic	---	bus_1_5	0.4	1	0.97717877
0	7	basic	---	bus_1_6	0.4	1	0.97640292
0	8	basic	---	bus_2_1	0.4	1	0.98938788
0	9	basic	---	bus_2_2	0.4	1	0.98775309
0	0	pv	---	Trafostatio	10	1	0.9863354
0	1	pv	---	main_busb	0.4	1	0
0	2	pv	---	bus_1_1	0.4	1	0
0	3	pv	---	bus_1_2	0.4	1	0
0	4	pv	---	bus_1_3	0.4	1	0
0	5	pv	---	bus_1_4	0.4	1	0
0	6	pv	---	bus_1_5	0.4	1	0
0	7	pv	---	bus_1_6	0.4	1	0
1	0	pv	---	Trafostatio	10	1	1
1	1	pv	---	main_busb	0.4	1	0.99178935
1	2	pv	---	bus_1_1	0.4	1	0.98662793
1	3	pv	---	bus_1_2	0.4	1	0.98232016
1	4	pv	---	bus_1_3	0.4	1	0.97886961
1	5	pv	---	bus_1_4	0.4	1	0.97627915
1	6	pv	---	bus_1_5	0.4	1	0.97455097
1	7	pv	---	bus_1_6	0.4	1	0.97368652
2	0	pv	---	Trafostatio	10	1	1
2	1	pv	---	main_busb	0.4	1	0.99178935
2	2	pv	---	bus_1_1	0.4	1	0.98662793

Analysis: Selecting Data (def anal_info)

Time Step	Bus Index	Scenario Name	Zone	Name	Voltage Level [kV]	In Service	Voltage [p.u]
step	index	scenario	zone	name	vn_kv	in_service	vm_pu
0	0	basic	---	Trafostatio	10	1	1
0	1	basic	---	main_busb	0.4	1	0.99265488
0	2	basic	---	bus_1_1	0.4	1	0.98802016
0	3	basic	---	bus_1_2	0.4	1	0.9841526
0	4	basic	---	bus_1_3	0.4	1	0.98105507
0	5	basic	---	bus_1_4	0.4	1	0.97872987
0	6	basic	---	bus_1_5	0.4	1	0.97717877
0	7	basic	---	bus_1_6	0.4	1	0.97640292
0	8	basic	---	bus_2_1	0.4	1	0.98938788
0	9	basic	---	bus_2_2	0.4	1	0.98775309
0	0	pv	---	Trafostatio	10	1	0.9863354
0	1	pv	---	main_busb	0.4	1	0
0	2	pv	---	bus_1_1	0.4	1	0
0	3	pv	---	bus_1_2	0.4	1	0
0	4	pv	---	bus_1_3	0.4	1	0
0	5	pv	---	bus_1_4	0.4	1	0
0	6	pv	---	bus_1_5	0.4	1	0
0	7	pv	---	bus_1_6	0.4	1	0
1	0	pv	---	Trafostatio	10	1	1
1	1	pv	---	main_busb	0.4	1	0.99178935
1	2	pv	---	bus_1_1	0.4	1	0.98662793
1	3	pv	---	bus_1_2	0.4	1	0.98232016
1	4	pv	---	bus_1_3	0.4	1	0.97886961
1	5	pv	---	bus_1_4	0.4	1	0.97627915
1	6	pv	---	bus_1_5	0.4	1	0.97455097
1	7	pv	---	bus_1_6	0.4	1	0.97368652
2	0	pv	---	Trafostatio	10	1	1
2	1	pv	---	main_busb	0.4	1	0.99178935
2	2	pv	---	bus_1_1	0.4	1	0.98662793

Data Frame

Index	step	index	scenario	vm_pu
0	0	0	basic	1
1	0	1	basic	0.992655
2	0	2	basic	0.98802
3	0	3	basic	0.984153
4	0	4	basic	0.981055
5	0	5	basic	0.97873
6	0	6	basic	0.977179
7	0	7	basic	0.976403
8	0	8	basic	0.989388
9	0	9	basic	0.987753
10	0	0	pv	0.986335
11	0	1	pv	0
12	0	2	pv	0
13	0	3	pv	0

Analysis: Filtering Data (def anal_vol)

Data Frame

Index	step	index	scenario	vm pu
0	0	0	basic	1
1	0	1	basic	0.992655
2	0	2	basic	0.98802
3	0	3	basic	0.984153
4	0	4	basic	0.981055
5	0	5	basic	0.97873
6	0	6	basic	0.977179
7	0	7	basic	0.976403
8	0	8	basic	0.989388
9	0	9	basic	0.987753
10	0	0	pv	0.986335
11	0	1	pv	0
12	0	2	pv	0
13	0	3	pv	0

In case : Violation

Index	Time Step	Bus Index	Scenario	Under Voltage [p.u]
0	0	5	basic	0.97873
1	0	6	basic	0.977179
2	0	7	basic	0.976403
3	0	1	pv	0
4	0	2	pv	0
5	0	3	pv	0
6	0	4	pv	0
7	0	5	pv	0
8	0	6	pv	0
9	0	7	pv	0
10	1	4	pv	0.97887
11	1	5	pv	0.976279
12	1	6	pv	0.974551
13	1	7	pv	0.973687

In case : No Violation

Index	Time Step	Line Index	Scenario	Percent
0	---	---	---	---

Default Violation Value

- Undervoltage : 0.98[p.u]
- Overvoltage : 1.02[p.u]

- Line overloading = 100[%]

- Transformer overloading = 100[%]

```
Analyze the output grid
Analyzing Voltage of the Bus
100%|          | 70120/70120 [07:26<00:00, 156.99it/s]
Analyzing Loading of the Lines
100%|          | 70112/70112 [00:20<00:00, 3353.19it/s]
Analyzing Loading of the Transformer
100%|          | 8764/8764 [00:18<00:00, 476.28it/s]
You can find Analysis information on Summary Tab
```

Analysis: Updating the Data (def anal_excel)

	Time Step	Bus Index	Scenario	Under Voltage [p.u]	Time Step	Line Index	Scenario	Line Loading Percentage[%]	Time Step	Trafo Index	Scenario	Transformer Loading Percentage[%]
0	0	5	basic	0.9787	---	---	---	---	---	---	---	---
1	0	6	basic	0.9772								
2	0	7	basic	0.9764								
3	0	1	pv	0								
4	0	2	pv	0								
5	0	3	pv	0								
6	0	4	pv	0								
7	0	5	pv	0								
8	0	6	pv	0								
9	0	7	pv	0								
10	1	4	pv	0.9789								
11	1	5	pv	0.9763								
12	1	6	pv	0.9746								
13	1	7	pv	0.9737								
14	2	4	pv	0.9789								
15	2	5	pv	0.9763								
16	2	6	pv	0.9746								
17	2	7	pv	0.9737								
18	3	4	pv	0.9789								
19	3	5	pv	0.9763								
20	3	6	pv	0.9746								
21	3	7	pv	0.9737								

- Data is updated into Summary tab
- In case of Violation
  : violation data will be written
- In case of No-Violation
  : ‘---’ will be written
- Existing Macro/VBA remain untouched
- Adding more violation will be added in another column
- Row information
  : Time Step / Index of the Component / Scenario / Value of the Violation

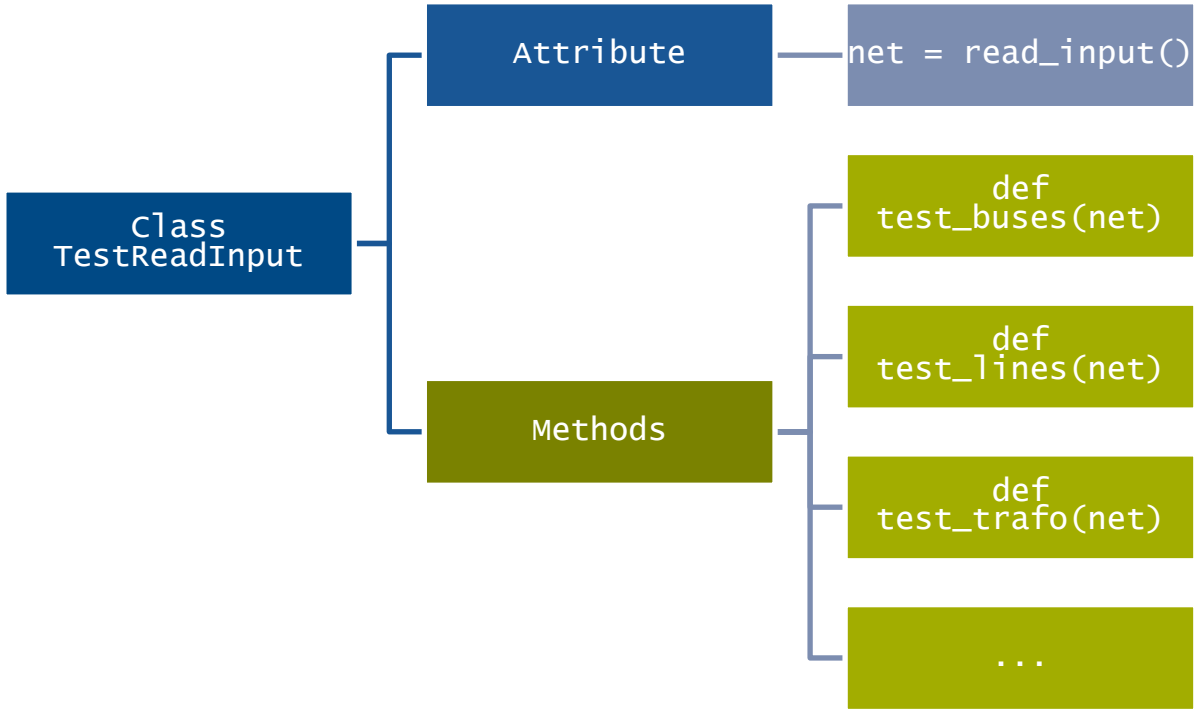
Analysis: Updating the Data (def anal_excel)

	Time Step	Bus Index	Under Voltage [p.u]	Time Step	Line Index	Loading Percent[%]	Time Step	Trafo Index	Loading Percent[%]
0	0	5	0.9787	1	0	26.2184	0	0	65.6994
1	0	6	0.9772	2	0	26.2184	1	0	65.6994
2	0	7	0.9764	3	0	26.2184	2	0	65.6994
3	0	1	0	4	0	26.2184	3	0	65.6994
4	0	2	0	5	0	26.2184	4	0	65.6994
5	0	3	0	6	0	26.2184	5	0	65.6994
6	0	4	0	7	0	26.2184	6	0	65.6994
7	0	5	0	8	0	26.1007	7	0	65.6994
8	0	6	0	9	0	25.1167	8	0	65.407
9	0	7	0	15	0	25.0901	16	0	65.6888
10	1	4	0.9789	16	0	26.2142	17	0	65.6994
11	1	5	0.9763	17	0	26.2184	18	0	65.6994
12	1	6	0.9746	18	0	26.2184	19	0	65.6994
13	1	7	0.9737	19	0	26.2184	20	0	65.6994
14	2	4	0.9789	20	0	26.2184	21	0	65.6994
15	2	5	0.9763	21	0	26.2184	22	0	65.6994
16	2	6	0.9746	22	0	26.2184	23	0	65.6994
17	2	7	0.9737	23	0	26.2184	24	0	65.6994
18	3	4	0.9789	24	0	26.2184	25	0	65.6994
19	3	5	0.9763	25	0	26.2184	26	0	65.6994
20	3	6	0.9746	26	0	26.2184	27	0	65.6994
21	3	7	0.9737	27	0	26.2184	28	0	65.6994
22	4	4	0.9789	28	0	26.2184	29	0	65.6994
23	4	5	0.9763	29	0	26.2184	30	0	65.6994
24	4	6	0.9746	30	0	26.2184	31	0	65.6994
25	4	7	0.9737	31	0	26.2184	32	0	64.9971
26	5	4	0.9789	32	0	25.9356	39	0	64.676
27	5	5	0.9763	39	0	25.8063	40	0	65.6994
28	5	6	0.9746	40	0	26.2184	41	0	65.6994
29	5	7	0.9737	41	0	26.2184	42	0	65.6994
30	6	4	0.9789	42	0	26.2184	43	0	65.6994
31	6	5	0.9763	43	0	26.2184	44	0	65.6994
32	6	6	0.9746	44	0	26.2184	45	0	65.6994
33	6	7	0.9737	45	0	26.2184	46	0	65.6994
34	7	4	0.9789	46	0	26.2184	47	0	65.6994

- Checking in case of changing the standard value of violation
  - Line Loading: 25 [%]
  - Transformer Loading: 64 [%]

Testing & Error Management

Testing: Unit Test

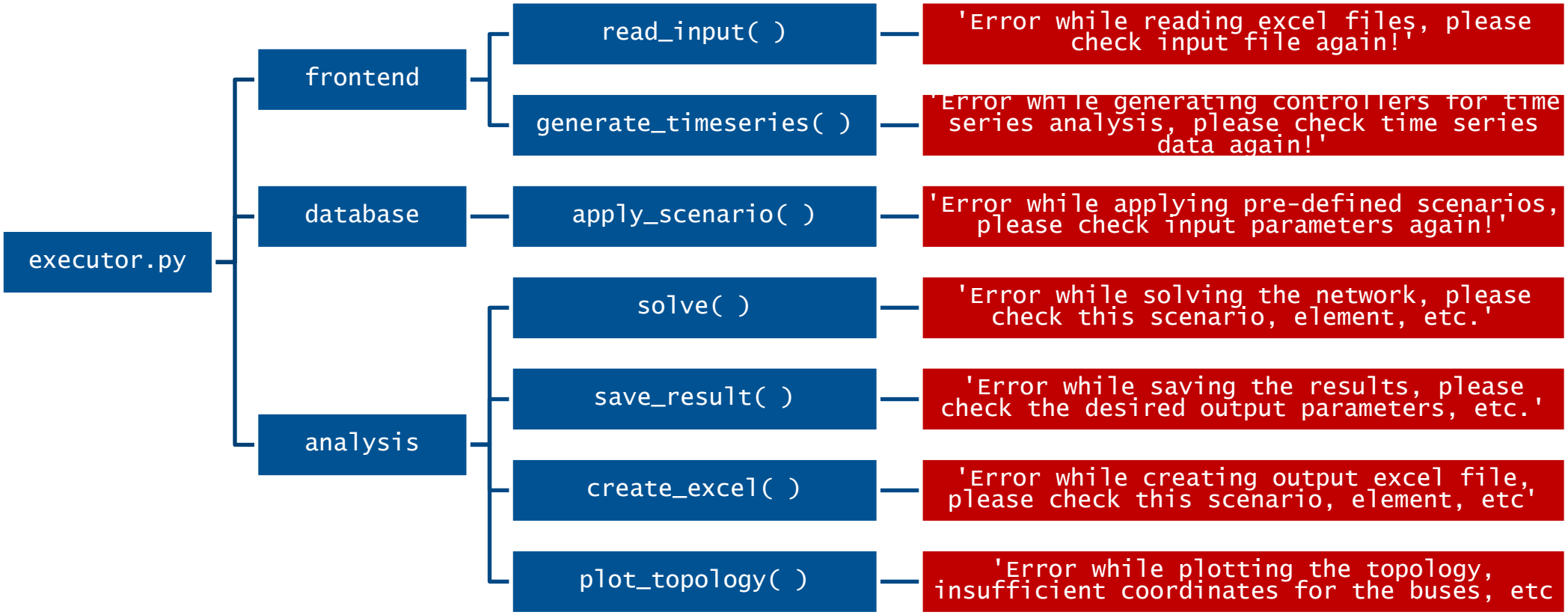


Example structure of test class

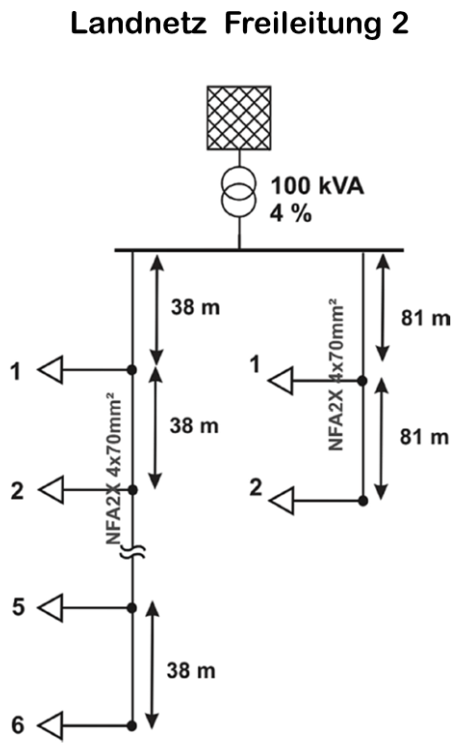
Module	statements	missing	excluded	coverage
src\analysis\excel_output.py	72	9	0	88%
src\analysis\parameters.py	85	19	0	78%
src\analysis\plot.py	25	17	0	32%
src\analysis\save.py	51	5	0	90%
src\analysis\solver.py	42	8	0	81%
src\analysis\time_series_func.py	42	12	0	71%
src\frontend\generate_timeseries.py	33	1	0	97%
src\frontend\read_input.py	19	0	0	100%
src\scenarios\apply_scenario.py	6	0	0	100%
src\scenarios\scenarios.py	45	9	0	80%
test\test_excel_output.py	36	1	0	97%
test\test_generate_timeseries.py	14	1	0	93%
test\test_parameters.py	42	1	0	98%
test\test_read_input.py	52	1	0	98%
test\test_save.py	59	1	0	98%
test\test_scenarios.py	41	1	0	98%
test\test_solver.py	90	1	0	99%
test\test_time_series_func.py	41	1	0	98%
Total	795	88	0	89%

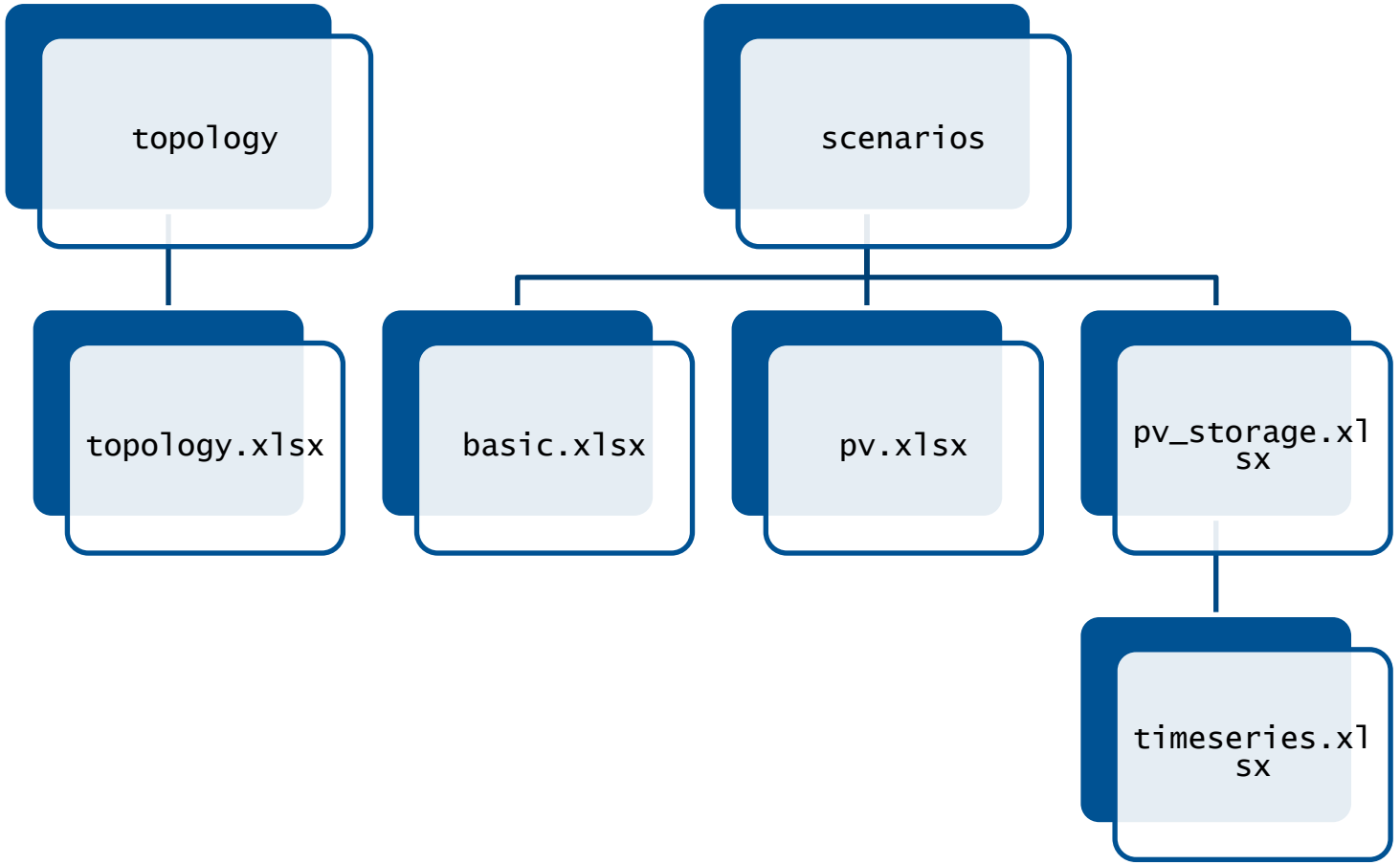
Test coverage

Error Management: System Overview



Example: Kerber Landnetze Freileitung 2





Check out our GitHub Repository: