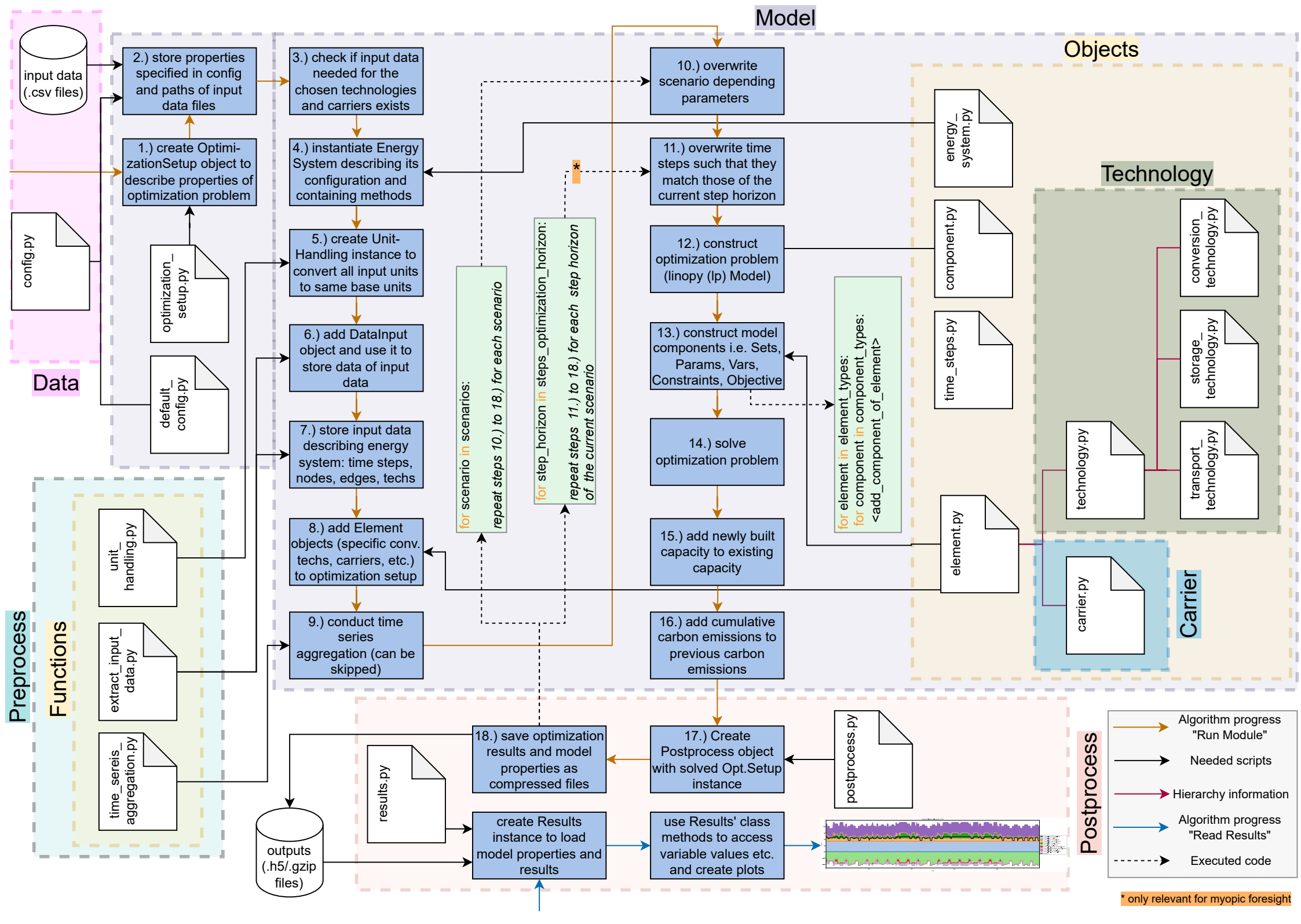
Model
Objects
Technology
Carrier
Data
Preprocess
Functions
Postprocess
input data (.csv files)
config.py
default_ config.py
optimization_ setup.py
2.) store properties specified in config and paths of input data files
1.) create OptimizationSetup object to describe properties of optimization problem
3.) check if input data needed for the chosen technologies and carriers exists
4.) instantiate Energy System describing its configuration and containing methods
5.) create UnitHandling instance to convert all input units to same base units
6.) add DataInput object and use it to store data of input data
7.) store input data describing energy system: time steps, nodes, edges, techs
8.) add Element objects (specific conv. techs, carriers, etc.) to optimization setup
9.) conduct time series aggregation (can be skipped)
10.) overwrite scenario depending parameters
11.) overwrite time steps such that they match those of the current step horizon
12.) construct optimization problem (linopy (lp) Model)
13.) construct model components i.e. Sets, Params, Vars, Constraints, Objective
14.) solve optimization problem
15.) add newly built capacity to existing capacity
16.) add cumulative carbon emissions to previous carbon emissions
17.) Create Postprocess object with solved Opt.Setup instance
18.) save optimization results and model properties as compressed files
create Results instance to load model properties and results
use Results' class methods to access variable values etc. and create plots
for scenario in scenarios: repeat steps 10.) to 18.) for each scenario
for step_horizon in steps_optimization_horizon: repeat steps 11.) to 18.) for each step horizon of the current scenario
*
for element in element_types: for component in component_types: <add_component_of_element>
energy_ system.py
component.py
time_steps.py
element.py
conversion_ technology.py
storage_ technology.py
technology.py
transport_ technology.py
carrier.py
unit_ handling.py
extract_input_ data.py
time_sereis_ aggregation.py
results.py
postprocess.py
outputs (.h5/.gzip files)
Algorithm progress "Run Module"
Needed scripts
Hierarchy information
Algorithm progress "Read Results"
Executed code
* only relevant for myopic foresight