



General structure

version 2.0 Feb 10, 2025

1 Plan4res directories

This document presents the structure of the plan4res directories, once installed.

Plan4res is distributed together with a container¹. The plan4res container, p4r-env, can be installed in various environments (Linux, Windows, MacOS). It provides a complete Linux environment with all dependencies already installed, allowing to run all plan4res modules. When installing plan4res (using the plan4res_install.sh script in [plan4res/install](#)) in a chosen directory (say, P4R_INSTALL_DIR), you will get the structure shown in Figure 1, and described in section 1.1.

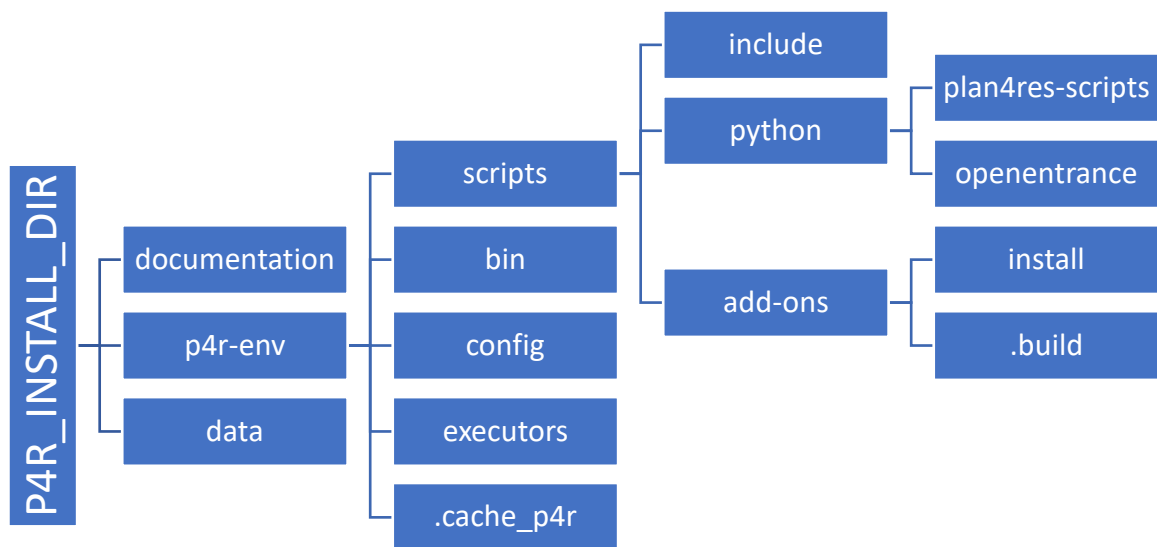


Figure 1: structure of P4R_INSTALL_DIR

You may also choose to run plan4res in a separated directory. This allows, in particular when installing on a server, to install the software only once, and have each user specify a personal directory (by using the user_init_plan4res.sh script in [plan4res/install](#)). Each user will then have in their P4R_LOCAL_DIR directory the structure shown in Figure 2:

¹ [How to explain containers in plain English | The Enterprisers Project](#)

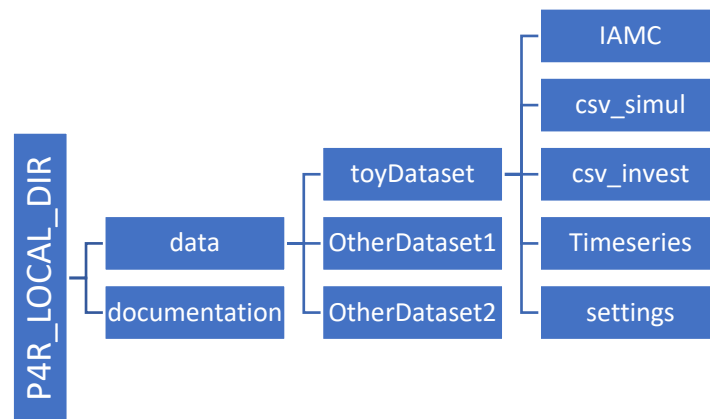


Figure 2: structure of P4R_LOCAL_DIR

1.1 P4R_INSTALL_DIR

This directory is structured as follows:

- Documentation: contains the plan4res documentation, downloaded from [plan4res/documentation: plan4res documentation](#)
- p4r-env : contains the plan4res environment (container) as well as all the modules of plan4res.
 - .cache_dir: plan4res container image (.SIF). In some cases it is necessary for the user to move the SIF files in another location as some systems specify mandatory locations for SIF images. In that case, it is necessary to update the p4r-env environment variable SINGULARITY_BIND with the location of the SIF file.
 - bin: contains the main p4r-env executables (bin to run the container and the executables used to update or rebuild the container)
 - config: contains the configuration file of p4r-env
 - executors: contains the definitions files of the container
 - scripts: contains all the plan4res modules
 - include: contains all plan4res bash scripts which are used to run the different modules
 - python:
 - plan4res-scripts: contains the python scripts for the CREATE, FORMAT and POSTTREAT modules
 - openentrance: contains the definition of the IAMC nomenclature such as developed in the open ENTRANCE² project ([openENTRANCE/openentrance: Definitions of common terms \(variables, regions, etc.\) for the openENTRANCE project](#))
 - add-ons: contains the solving modules of plan4res: SSV, SIM and CEM, which are based on the SMS++³ library.
 - Install: installation dir for sms++ and the solvers used within sms++
 - .build: directory where the source code is downloaded and compiled

² [openENTRANCE – open ENergy TRanstion ANalyses for a low-Carbon Economy](#)

³ [SMS++ / The SMS++ Project · GitLab](#)

- data: optional directory. Created only when the user wishes to run plan4res in the same location where it is installed. The structure of data is described in section 1.3

1.2 P4R_LOCAL_DIR

This directory is structured as follows:

- Documentation: contains the plan4res documentation, downloaded from [plan4res/documentation: plan4res documentation](https://plan4res.github.io/)
- data : directory where the user will run plan4res. The structure of data is described in section 1.3

1.3 Dataset directory

Figure 3 shows the structure of plan4res.

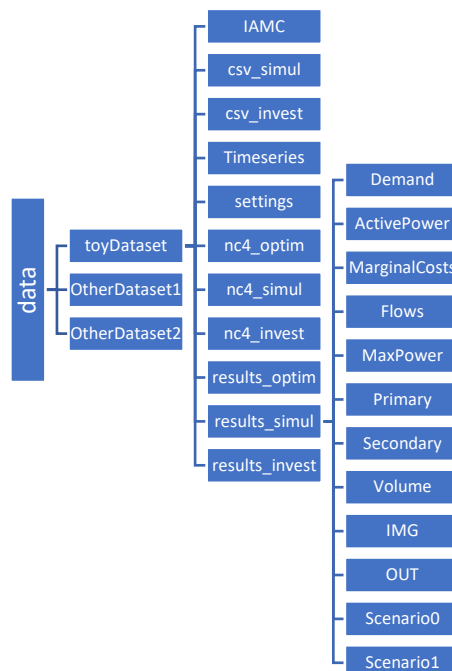


Figure 3: structure of data

Each dataset corresponds to a directory in data (here toyDataset, Otherdataset1 and OtherDataset2). Datasets always have the same structure:

- Timeseries: contains the timeseries which are used as inputs, as well as the timeseries created by the data treatments of plan4res – the CREATE module- (aggregated timeseries over regions, or timeseries with a different time granularity)
- IAMC: contains the scenario which can be used to create the plan4res input data (this can be a result of an energy modelling tool such as e.g. GENeSYS-MOD. The dataset used must be a csv file named DATASE.csv. It must comply to the IAMC format⁴.
- csv_simul and csv_invest: those 2 sub repositories contain the native input files of plan4res. These can be created manually by the user or from a scenario in IAMC/ by using the CREATE module of plan4res. csv_simul contains files which cannot be used for performing an

⁴ Description of the IAMC Format: <https://doi.org/10.5281/zenodo.5521098>

investment study, while csv_invest contains files which can be used for performing an investment study. The difference between those 2 sets of files lies in the fact that csv_invest may include some generation assets which are described only because it is allowed to invest in these units, while their capacity is 0. As the investment module (CEM) can only increase the existing capacities, it is necessary to start from a dataset with all possible technologies. The 'new' technologies are then present but with a very low capacity (whose level is defined in the plan4res settings files).

- nc4_optim, nc4_simul and nc4_invest: these 3 sub repositories contain the plan4res NetCDF inputs, which are created from the data in csv_XXX/ and in Timeseries/ by the FORMAT module of plan4res. These datasets are not easily readable so the user may not look at them.
- results_optim: This repo will contain the results of the SSV module
- results_simul: This repo will contain the results of the SIM module
- results_invest: This repo will contain the results of the CEM module
- settings: This repo contains all the configuration files used by all different modules of plan4res.

The data generation modules CREATE and FORMAT as well as the solving modules SSV, SIM and CEM also create the directories when they are not yet present. It is then possible to run CREATE when only IAMC and Timeseries are present, or FORMAT when only Timeseries and csv_simul are present.

2 Plan4res modules and workflow

plan4res is composed of the modules listed below. The workflow between the modules is illustrated in Figure 4.

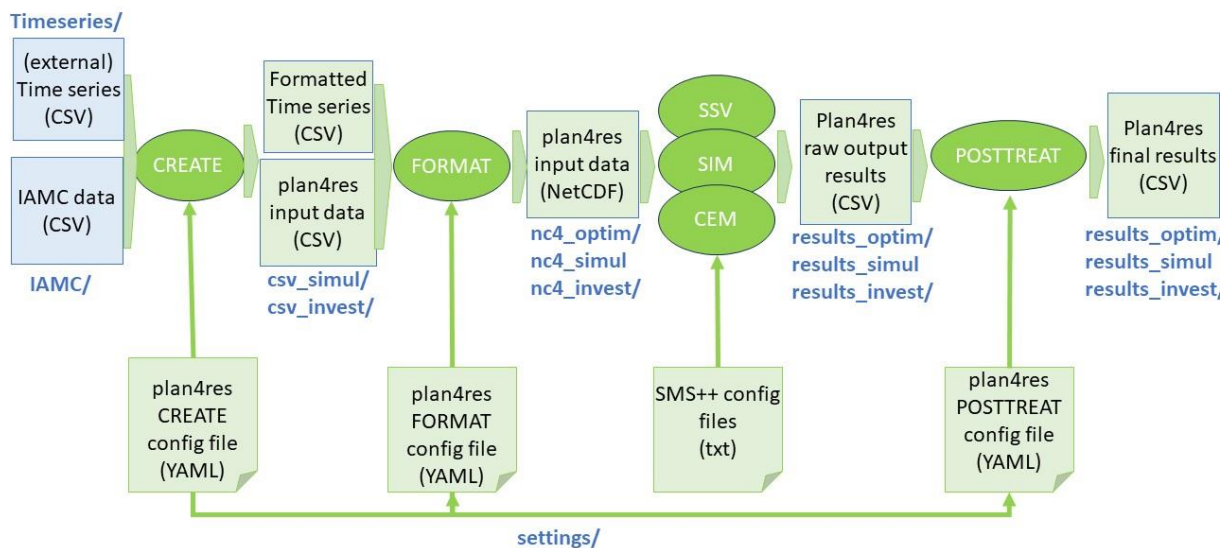


Figure 4: plan4res workflow

- CREATE: creates a plan4res CSV dataset out of a scenario in IAMC format (in IAMC/) and hourly timeseries
- FORMAT: creates a plan4res NetCDF dataset out of the plan4res CSV input data and the hourly timeseries
- SSV: solves the Seasonal Storage Valuation problem. Uses NetCDF files in nc4_optim and computes Bellman Values, stores the results in results_optim

- SSV: solves the Seasonal Storage Valuation problem. Uses NetCDF files in nc4_optim and computes Bellman Values, stores the results in results_optim
- SIM: solves the simulation problem. Uses NetCDF files in nc4_simul and computes schedules of all assets and marginal costs, stores the results in results_simul
- CEM: solves the capacity expansion problem. Uses NetCDF files in nc4_invest and computes new investments, as well as the schedules of all assets and marginal costs for the new power system, including invested assets, stores the results in results_invest