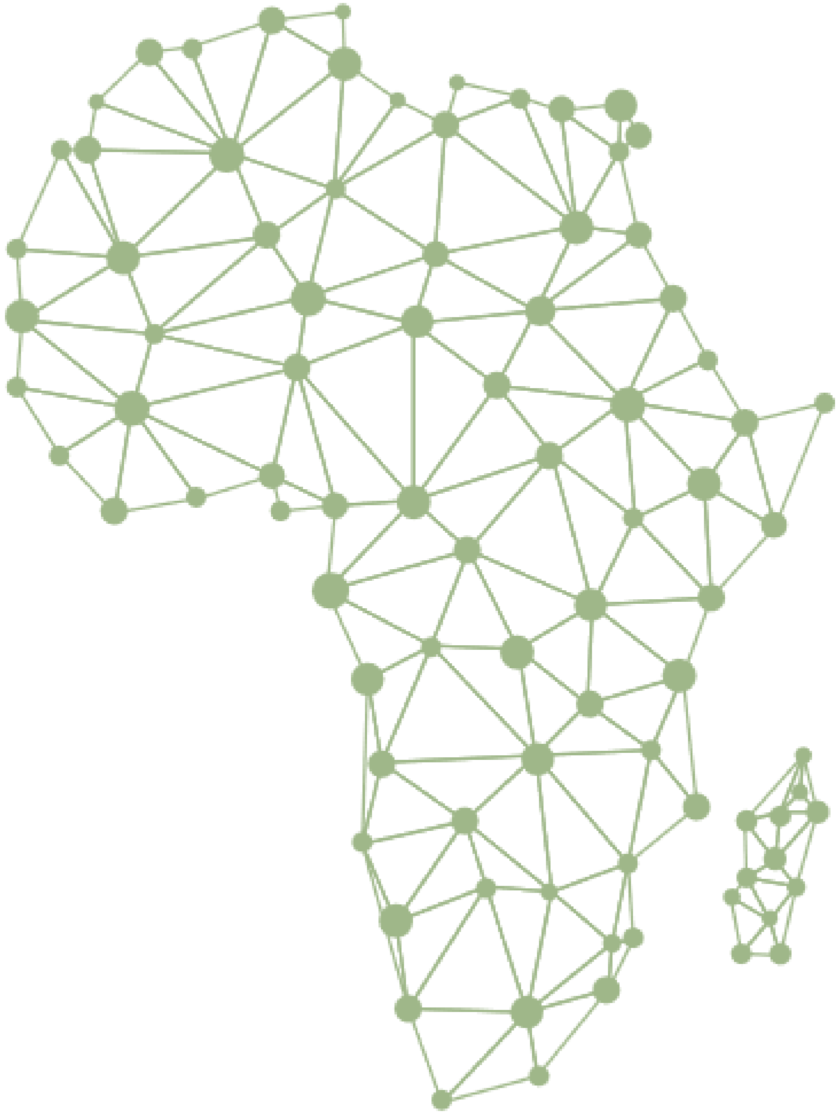


OpenMod Africa

plan4res

plan4res workshop Madrid

Sandrine Charousset,
EDF

June, 19, 2024

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

OpenMod Africa

plan4res

The model

This project receives funding from the European Union's Ho

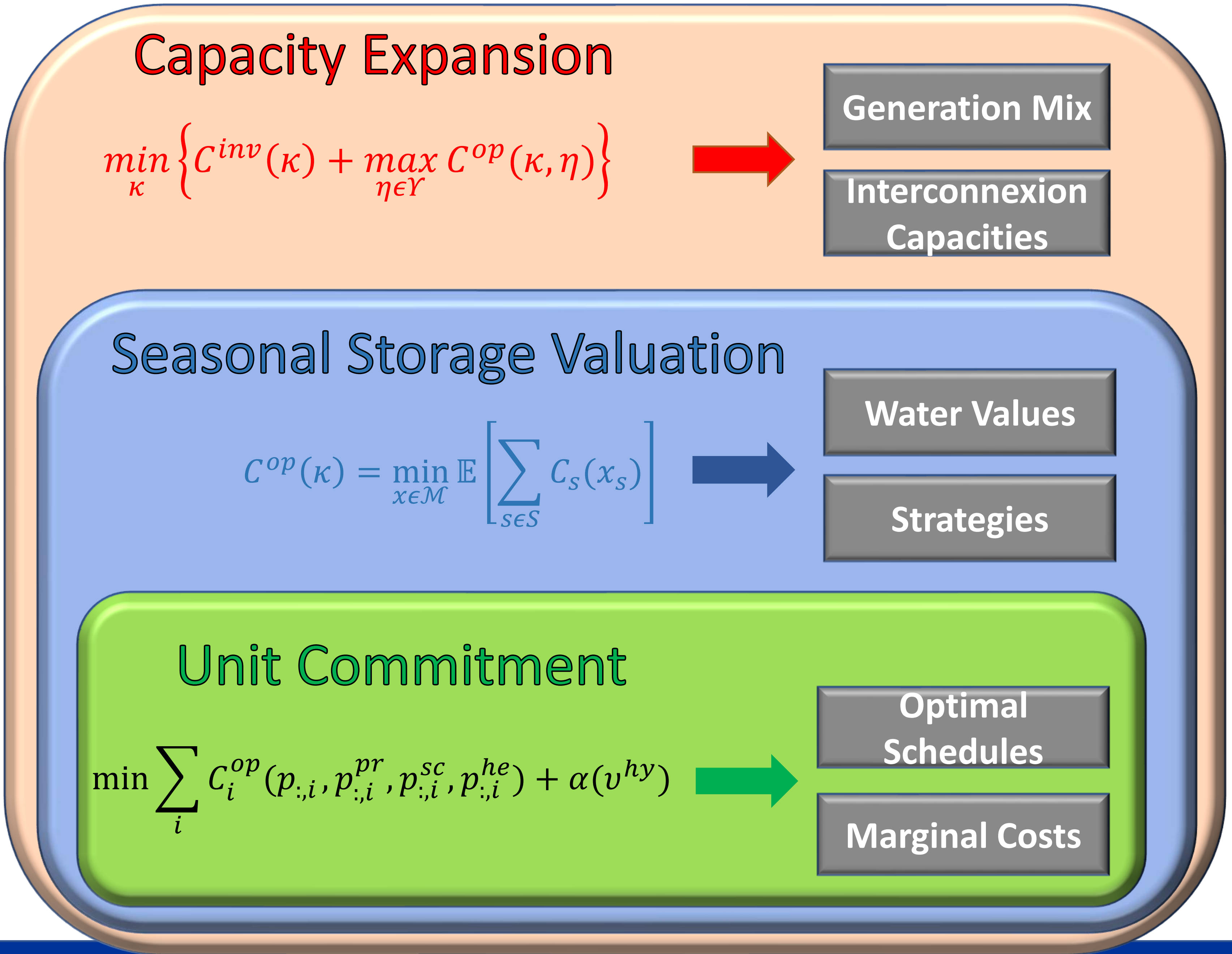
The plan4res electricity system modelling suite

a Stochastic Power System model composed of 3 embedded layers:

- ***The Capacity expansion model computes the optimal mix on a given year***
  - ✓ electric generation plants,
  - ✓ Short term storages (batteries….),
  - ✓ interconnection capacities
- ***The seasonal storage valuation model computes the optimal strategy for seasonal storages***
  - ✓ For Hydro reservoirs
  - ✓ And also all other ‘seasonal’ flexibilities such as Seasonal Demand response
- ***The European unit commitment (EUC) model computes the optimal dispatch:***
  - ✓ Supply power demand and ancillary services
  - ✓ Minimal inertia in the system
  - ✓ Maximum transmission and distribution capacities between clusters
  - ✓ Technical (including dynamic) constraints of all assets

This project receives funding from the European Union’s Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 1011181

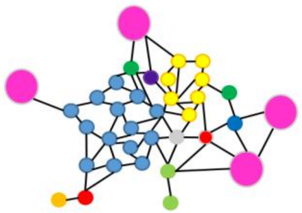
- **The Capacity expansion model** computes the optimal mix:
  - ✓ *electric generation plants,*
  - ✓ *storages,*
  - ✓ *interconnection capacities between clusters*
  - ✓ *distribution grid capacities,*
- **The seasonal storage valuation model** computes the operation strategies for seasonal storages:
  - ✓ *For Hydro reservoirs*
  - ✓ *And also all other 'seasonal' flexibilities such as Demand response*
- **The unit commitment model** computes the optimal operation schedule for all the assets dealing with constraints:
  - ✓ *Supply power demand and ancillary services*
  - ✓ *Minimal inertia in the system*
  - ✓ *Maximum transmission and distribution capacities between clusters*
  - ✓ *Technical constraints of all assets*



m the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Main characteristics

- ## Adaptable Geography perimeter
  - Europe or lower perimeter
  - Subcountry representation is possible
- ## Uncertainties:
  - Electricity demand
  - RES profiles (PV, Wind, RoR…)
  - Inflows
  - Failures of traditional plants
- ## Modular Time horizon and granularity
  - Typically 1 yr. with hourly granularity
- ## Modular Grid
  - Regions and interconnections
  - Regions can be:
    - Countries
    - Groups of countries
    - Sub country regions



❑ Power plants

- Operational decision of power plants based on their specific fuel costs
- Technical constraints (ramping, min up-/downtimes,…)

❑ Storages

- Hydro storages including complex cascaded systems
- Battery storages

❑ Intermittent generation

- Generation of wind, solar, run of river based on meteorological profiles

❑ E-mobility

- Storage capability of electric vehicles (vehicle-to-grid, power-to-vehicle)
- Limitation of storage availability by driving profiles

❑ Demand Response

- Load shifting of a given energy consumption during a sub-period
- Load curtailment based on a given potential (e.g. during one year)

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Inputs and Outputs

This project receives funding from the European Union's Ho

Inputs and Outputs

- Generation mix (capacities, costs, constraints) (incl. Storages....)
- Electricity demand and system services rqrts
- Uncertainties

- Investment costs
- Potentials for investment

plan4res Unit Commitment

- Generation schedules
- Marginal costs
- Generation cost

plan4res Seasonal Storage Valuation

- Bellman Values (= strategies for seasonal storage management)

plan4res Capacity Expansion

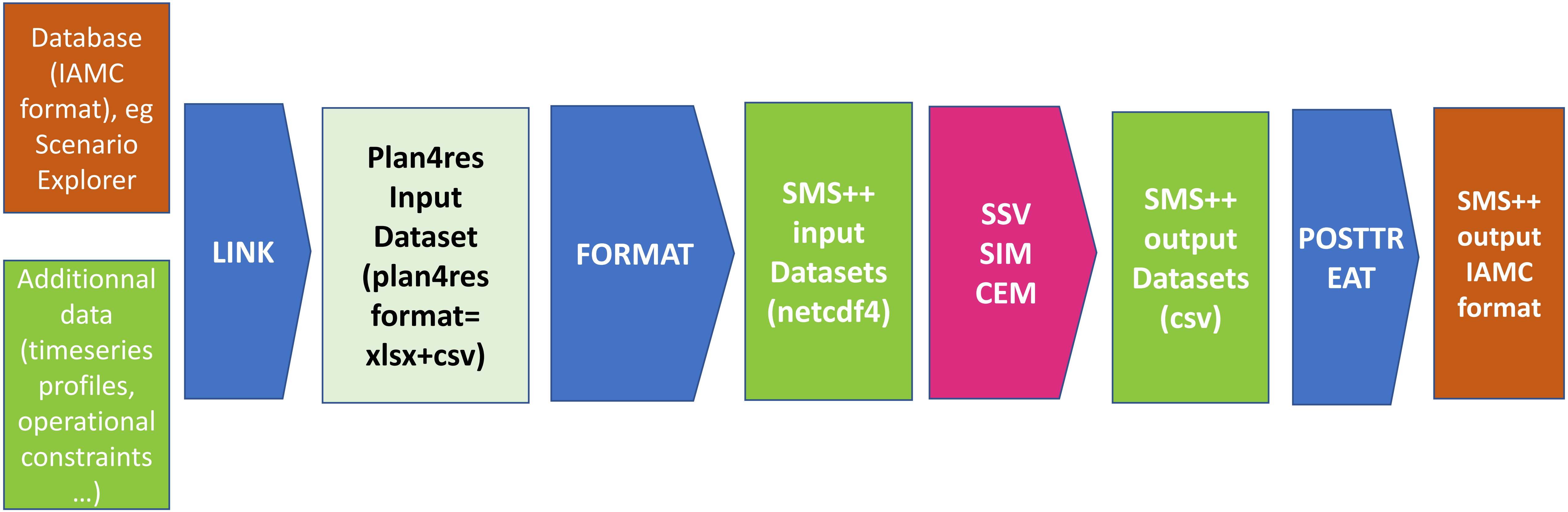
- Invested Generation capacities
- Invested Interconnection capacities
- Invested short term storage capacities

This project receives funding from the European Union’s Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Design of the tool

This project receives funding from the European Union's Ho

plan4res design



This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

plan4res design

- Plan4res is composed of:
  - Linkage scripts: from and to IAMC format (python)
  - Formatting tool:
    - Creates NetCDF4 SMS++ inputs
    - C++ version available in the container
    - Currently being rewritten in python
  - Plan4res Solvers
    - Written in C++ based on SMS++

https://gitlab.com/smspp/smspp-project

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation u

OpenMod Africa

plan4res

Main modules:

- ucblock_solver
- sddpsolver
- investmentsolver

SMS++

This project receives funding from the European Union's Ho

Plan4res solving modules

- 3 modules
  - *(ucblock_solver: solves a short term deterministic unit commitment problem (usually on 1 week))*
  - sddp_solver:
    - Compute bellman values
    - Solves a serie of unit commitments on a sequence of periods, with bellman values as input
  - Investment_solver
    - Compute optimal investments
    - Simulates (= solves series of unit commitment problems on all periods, for all scenarios)

https://gitlab.com/smspp/smspp-project

This project receives funding from the European Union’s Horizon 2023 Framework Programme for Research and Innovation u

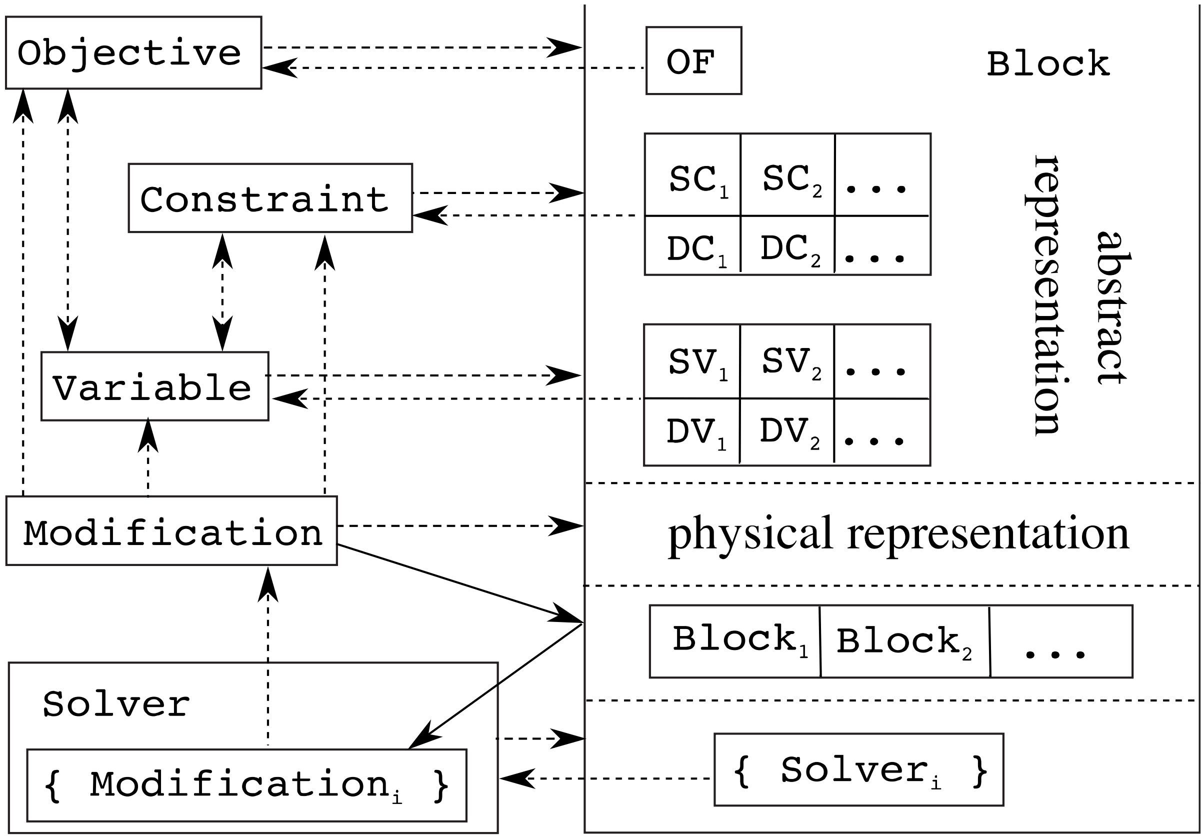
Plan4res solving modules

- The 3 modules are implemeted with the SMS++ framework (developped by University of Pisa)
- They are available within a container (developped by HPE)
- Can run on any platform (Unix, Windows, Mac), but more adapted to Linux
- Includes parallelisation

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation u

Modelling with SMS++

- SMS++ is a set of C++ classes implementing a modelling system that:
  - allows exploiting specialised solvers
  - manages all types of dynamic changes in the model
  - Explicitely handles reformulation/restriction/relaxation
  - does parallel from the start
  - should be able to deal with almost anything (bilevel, PDE,..)
  - ***Includes specialized blocks for energy system modelling***

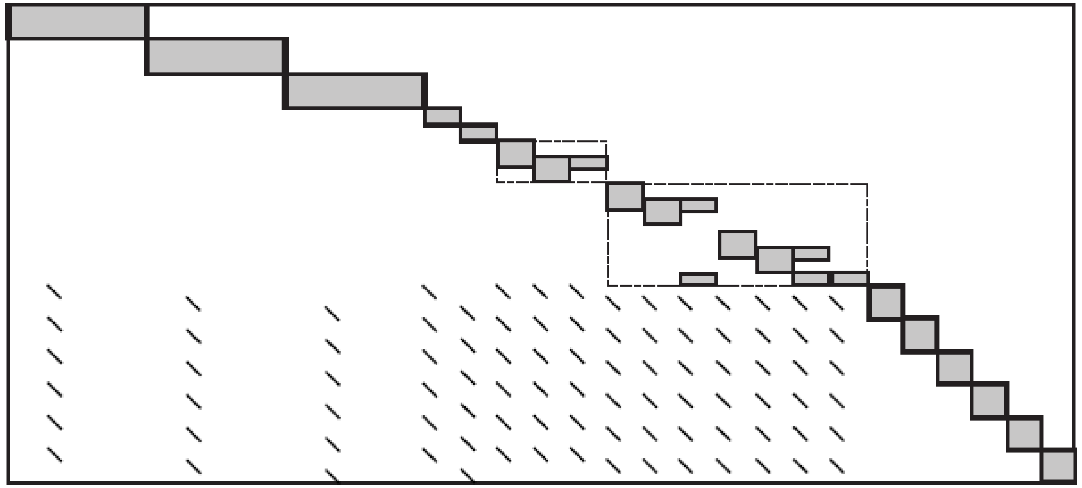


nework Programme for Research and Innovation no. 101118123

Modelling with SMS++

Nested decompositions at different time horizons

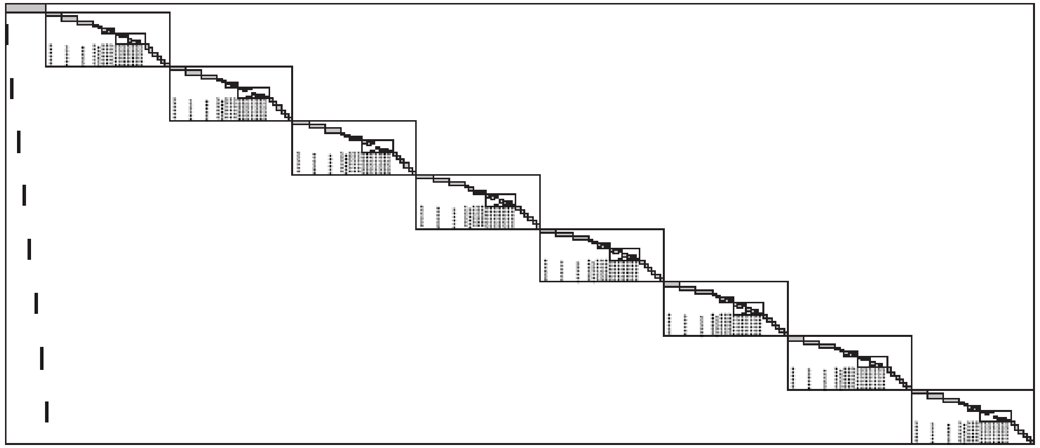
- Schedule a set of generating units to satisfy the demand at each node of the transmission network at each time instant of the horizon (24h)



- Several types of almost independent blocks + linking constraints
- Perfect structure for Lagrangian relaxation[1,2]

[1] Borghetti, F., Lacalandra, Nucci "Lagrangian Heuristics Based on Disaggregated Bundle Methods [...]", *IEEE TPWRS*, 2003
[2] Scuzziato, Finardi, F. "Comparing Spatial and Scenario Decomposition for Stochastic [...]" *IEEE Trans. Sust. En.*, 2018

- Manage water levels in reservoirs considering uncertainties (inflows, temperatures, demands, ...) to minimize costs over the time horizon



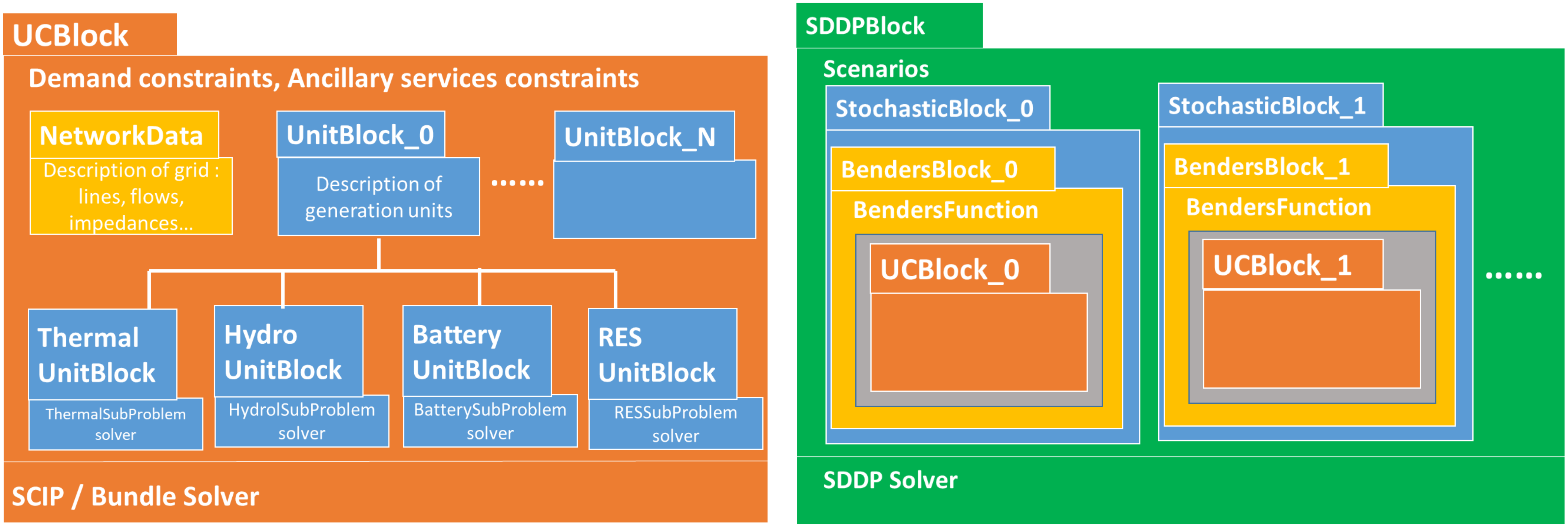
- Very large size, nested structure
- Perfect structure for Stochastic Dual Dynamic Programming[3,4] with multiple EUC inside

[3] Pereira, Pinto "Multi-stage stochastic optimization applied to energy planning" *Math. Prog.*, 1991
[4] van-Ackooij, Warin "On conditional cuts for Stochastic Dual Dynamic Programming" arXiv:1704.06205, 2017

Source: A. Frangioni, Uni Pisa

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

The Seasonal Storage Valuation and Unit Commitment in SMS++



Installation recap

This project receives funding from the European Union's Ho

Plan4res structure

plan4res is composed of the following pieces:

- **The p4r-env container :** https://gitlab.com/cerl/plan4res/p4r-env
- **The SMS++ modelling and optimization library :** https://gitlab.com/smspp
- **The plan4res python linkage, pre/post processing, visualisation scripts :** https://github.com/openENTRANCE/plan4res-scripts
- **Launching scripts, documentation, example of datasets :** https://github.com/openENTRANCE/plan4res

Installing plan4res requires installing each piece

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing p4r-env

p4r-env is the main container

It includes:

- A full linux installation (currently debian:bullseye)
- All dependences required by of SMS++ (in particular boost, eigen, netcdf-C++, see https://gitlab.com/smspp/smspp#getting-started)
- Python3 and all packages needed by the plan4res python scripts

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing p4r-env in windows

Requirements:

- Windows 7 pro 64bit SP1 or higher
- powershell 3.0 or higher
- CPU must support hardware virtualization (which may require beeing enabled in the BIOS)

Procedure:

- Install Git for Windows (use default settings) https://git-for-windows.github.io/
- Install VirtualBox and Extension Pack https://www.virtualbox.org/wiki/Downloads
- Install Vagrant https://www.vagrantup.com/downloads.html
- (Optional) Install Vagrant Manager http://vagrantmanager.com/downloads/

Vagrant and VirtualBox allow to emulate a UNIX system on the windows computer

See **https://gitlab.com/cerl/plan4res/p4r-env#windows**

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing p4r-env in windows

Commands for windows installation:

- Run Git Bash
- Within Git Bash:
    - git clone --recursive https://gitlab.com/cerl/plan4res/p4r-env
    - cd p4r-env
    - Edit Vagrantfile to give more memory….
    - git config submodule.recurse true
    - vagrant plugin install vagrant-proxyconf
    - vagrant up
    - vagrant halt

Creates structure p4r-env

Starts the container (first time downloads image)

Stops the container

You can set the RAM and CPU allocated to the VM by editing parameters `vb.cpus` and `vb.memory` in file p4r-env\Vagrantfile. We advise setting at least **4096 Mb of RAM**!

See https://gitlab.com/cerl/plan4res/p4r-env#windows

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing p4r-env in linux

Commands for linux installation:

- Create a directory (install_dir)
  - ➢ mkdir install_dir
- Download p4r-env:
  - ➢ git clone --recursive https://gitlab.com/cerl/plan4res/p4r-env
  - ➢ cd p4r-env
  - ➢ *(if you are using OpenMPI, edit config/plan4res.conf => change MPICH to OpenMPI – see next slide)*
  - ➢ bin/p4r
  - ➢ exit

Creates structure p4r-env

Starts the container (first time downloads image)

Stops the container

See **https://gitlab.com/cerl/plan4res/p4r-env#linux**

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Adaptations of p4r-env to local needs

If your system allows parallelisation

- Check mpi version :

  mpiexec –version

- 2 versions of the container are available depending on MPI installation: openMPI and MPICH

  Default version of the container is for MPICH

  To change to openMPI:

  - **edit file p4r-env/config/plan4res.conf**
  - **change value of P4R_MPI_IMP:**
    - **Instead of** **P4R_MPI_IMP=${P4R_MPI_IMP:-"MPICH"}**
    - **Write:** **P4R_MPI_IMP=${P4R_MPI_IMP:-“OpenMPI"}**

This project receives funding from the European Union’s Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Adaptations of p4r-env to local needs

To prevent download of SIF image each time you run bin/p4r (or any Launch)

- **edit file p4r-env/config/plan4res.conf**
- **change value of P4R_SINGULARITY_IMAGE_PRESERVE:**
  - **Instead of**

P4R_SINGULARITY_IMAGE_PRESERVE=${P4R_SINGULARITY_IMAGE_PRESERVE:-0}

  - **Write**

P4R_SINGULARITY_IMAGE_PRESERVE=1

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing SMS++ in p4r-env

Requirements:

- You must have a linux installer of CPLEX (even if installing on a windows machine!!)
  => cplex_studioXXXX.bin (XXXX depends on the version of CPLEX)

Procedure (for academics to get free version of CPLEX):

- Go to IBM ILOG CPLEX Optimization Studio: https://www.ibm.com/products/ilog-cplex-optimization-studio
- click "Try it free" =>You will be asked for create an account as an academic or use an already existing one, then you will be directed to the download page
- Download the LINUX version of the installer bin (cplex_studioXXX.bin)

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing SMS++ in p4r-env

For Windows users, if necessary:
Edit *install_dir\p4r-env\scripts\add-ons\sms++*: replace 3 instances of `make -j$(getconf _NPROCESSORS_ONLN)` with `make –j1`.

Commands :

- Commands are launched from the directory p4r-env
- Install StOpt (stochastic optimization library)
  - (bin/p4r add-on stopt uninstall)
  - bin/p4r add-on stopt
- Install SMS++
  - (bin/p4r add-on sms++ uninstall)
  - bin/p4r add-on sms++ CPLEX=<Your-CPLEX-Linux-Installer.bin>

Always install before SMS++

only if old install already exists

Install sms++ executables in p4r-env

See https://gitlab.com/cerl/plan4res/p4r-env#p4r-env

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Installing the python scripts in p4r-env

Commands (You are still located in the directory p4r-env)

- cd scripts
- mkdir python
- cd python
- git clone https://github.com/openENTRANCE/openentrance.git
- git clone https://github.com/openENTRANCE/plan4res-scripts.git

Install Open ENTRANCE nomenclature

Install linkage, pre/post-treatment and visualisation scripts

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Get documentation, config files and launch scripts

- From your install_DIR
- git clone https://github.com/openENTRANCE/plan4res.git
- Copy launch scripts plan4res/run* to p4r-env
- Copy the sub-dir plan4res/include to p4r-env/scripts/
- Copy the datasets in ExampleData to p4r-env/data/local/

Creates plan4res dir in Install_Dir, populated with…. (see next slide)

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Plan4res input data

This project receives funding from the European Union's Ho

What do we need ?

- to be specified in settingsCreateInputPlan4res_xx.yml

- Yearly data:
  - Electricity demand (Final Energy|Electricity)
  - Interconnections (Network|Electricity|Maximum Flow)
  - Installed Capacities (Capacity|Electricity|)
  - Costs (Variable Cost (incl. Fuel Cost)|Electricity|)
- Hourly Timeseries

This project receives funding from the European Union’s Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

The IAMC file as converted from GENeSYS-MOD inputs and outputs

Example in
https://github.com/openENTRANCE/plan4res/tree/main/ExampleData/mini/IAMC

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Plan4res datasets

- Main files:
  - Regions: ZP_ZonePartition
  - Demand: ZV_ZoneValues
  - Interconnections: IN_Interconnections
  - Thermal Power: TU_ThermalUnits
  - Hydro Réservoirs: SS_SeasonalStorage
  - Renewable Power: RES_RenewableUnits
  - Storage: STS_ShortTermStorage
- TimeSeries:
  - Demand Profiles
  - Inflows
  - Renewable load factors
  - …

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Example in
https://github.com/openENTRANCE/plan4res/tree/main/ExampleData/mini/csv_simul

https://github.com/openENTRANCE/plan4res/tree/main/ExampleData/mini/csv_invest

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Creating the Plan4res datasets

CreateInputPlan4res.py

runCREATE.sh

Usage: from p4r-env,

./runCREATE.sh NAMEDATASET simul → Dataset for simulation

or

./runCREATE.sh NAMEDATASET invest → Dataset for investment optimisation

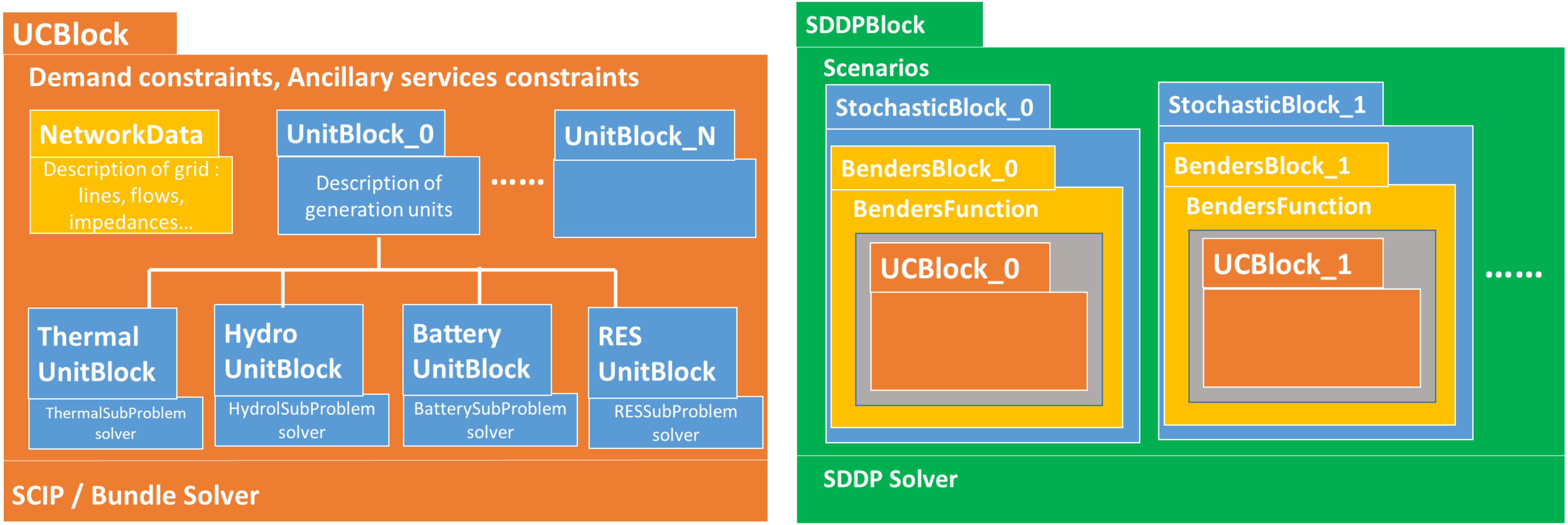
This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

SMS++ datasets

- Main files:
  - InvestmentBlock.nc4 → Describes the investment optimisation problem
  - SDDP_Block.nc4 → Describes the problem on 1 year
  - Block_$i.nc4 → Describes the short term problems and the power system

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

The Seasonal Storage Valuation and Unit Commitment in SMS++



Example in
https://github.com/openENTRANCE/plan4res/tree/main/ExampleData/mini/nc4_simul

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Creating the netCDF files

format.py

runFORMAT.sh

Usage: from p4r-env,

./runFORMAT.sh NAMEDATASET simul → Dataset for simulation

or

./runCREATE.sh NAMEDATASET optim → Dataset for computing bellman values

or

./runCREATE.sh NAMEDATASET invest → Dataset for investment optimisation

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Running the models

- 3 solvers available:
  - SDDP → Computes Bellman values
  - Simulation → Computes schedules of all units
  - Investment → Computes a more adapted power mix

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Running the SDDP

runSSV.sh

Usage: from p4r-env,

./runSSV.sh NAMEDATASET

BellmanValuesOUT

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Running the Simulation

runSIM.sh

Usage: from p4r-env,

./runSIM.sh NAMEDATASET

ActivePower
MarginalCosts
Flows
…..

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123

Running the Investment

runCEM.sh

Usage: from p4r-env,

./runCEL.sh NAMEDATASET

SolutionOUT.csv
And all results of simulation

This project receives funding from the European Union's Horizon 2023 Framework Programme for Research and Innovation under Grant Agreement no. 101118123