energyRt

energy systems modeling R-toolbox

Oleg Lugovoy
Vladimir Potashnikov

Open Energy Modelling Workshop
October 27-28, Milano 2016

Outline

- Demand for **reproducible research**

- The concept of **object oriented programming** (OOP)

- **energyRt**
    - language
    - main objects
    - main methods
    - Main features

Motivation

- minimize **time** of development and application of RES/BottomUp models,
- boost **learning curve** in energy modeling,
- improve **transparency** and understanding of energy models,
- use power of **open-source** to improve energy models and their application,
- making **reproducible research** accessible in RES-modeling,
- **integration** with other models and software.

General modeling workflow

- Model development
  - Design model structure
  - Data collection, processing
  - Model implementation (software)
  - Calibration of the model
- Model application
  - Design scenarios
  - Adjust, update model
  - Running the model
  - Processing results
  - Analysis of the results
  - Conclusions

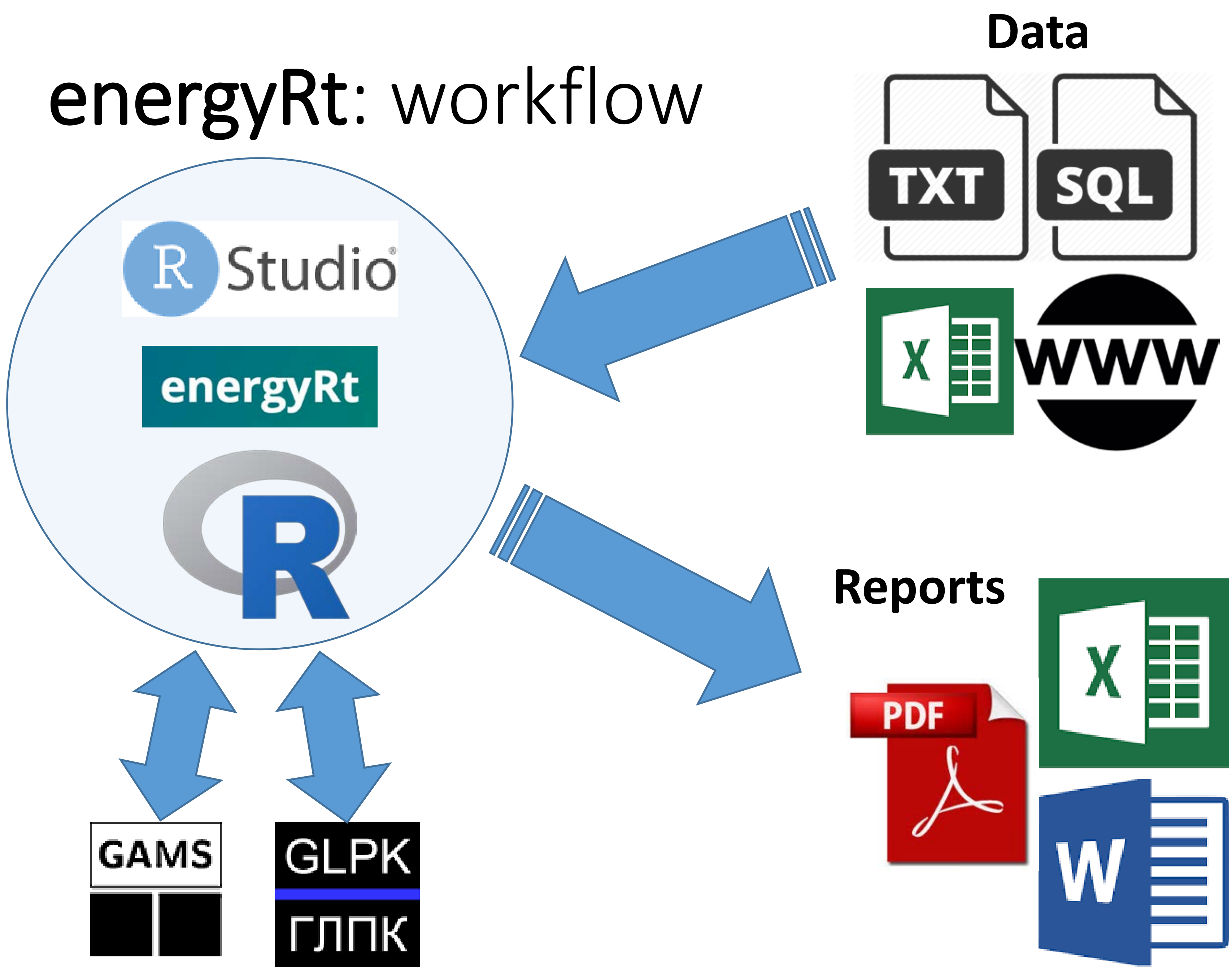
General modeling workflow

- Model development
  - Design model structure
  - Data collection, processing
  - Model implementation (software)
  - Calibration of the model
- Model application
  - Design scenarios
  - Adjust, update model
  - Running the model
  - Processing results
  - Analysis of the results
  - Conclusions

Demand for reproducible research in energy modeling: an example

- How many model runs have your team done to reach “reasonable” results?
- How much time have you and your team spent for processing your model results and finding inconsistencies?
- How much time will it take to reproduce your own results?
- Will be able other people (new team members) reproduce your research?

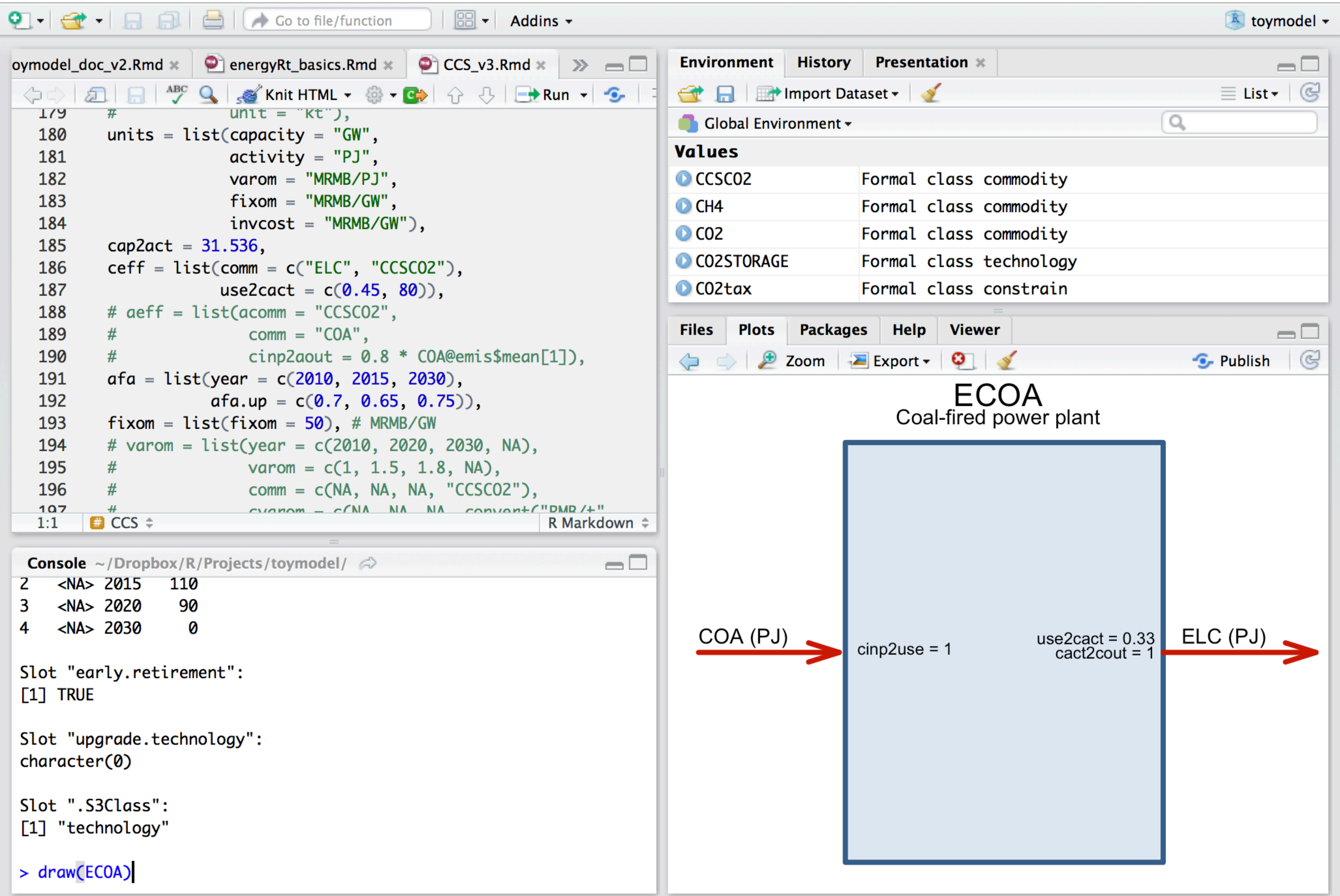
energyRt: workflow



General modeling workflow with **energyRt**

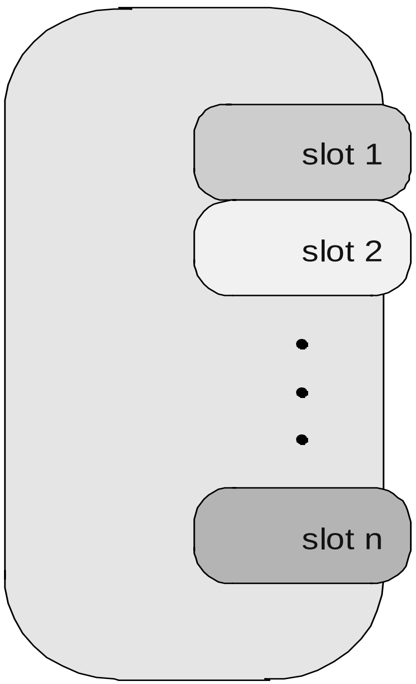
- Model development
  - Design model structure
  - Data collection, processing (R, energyRt classes: *commodity*, *technology*, *repository*)
  - Model implementation (energyRt classes: *model;* code: GAMS or GLPK)
  - Calibration of the model (energyRt methods: *draw*, *levcost*, *compare*, …)
- Model application
  - Design scenarios (energgRt classes: constrain, scenario)
  - Adjust, update model (R, energyRt)
  - Running the model (energyRt methods: *solve*)
  - Processing results (energyRt methods: report, *getData*, …)
  - Analysis of the results (R graphs; energyRt methods: *plot*, *report*, …)
  - Conclusions

R + RStudio + energyRt



Classes: custom data structure

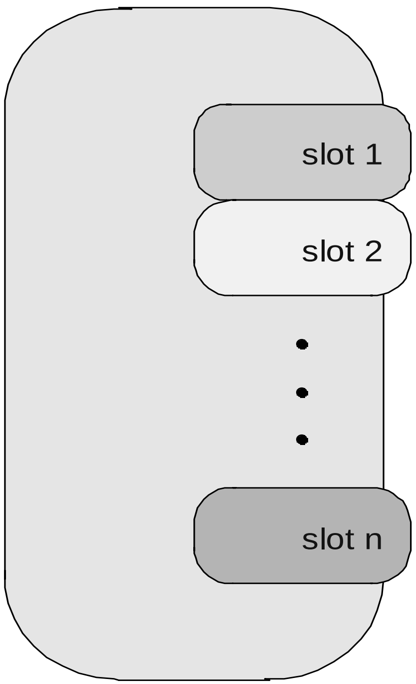
An object/class



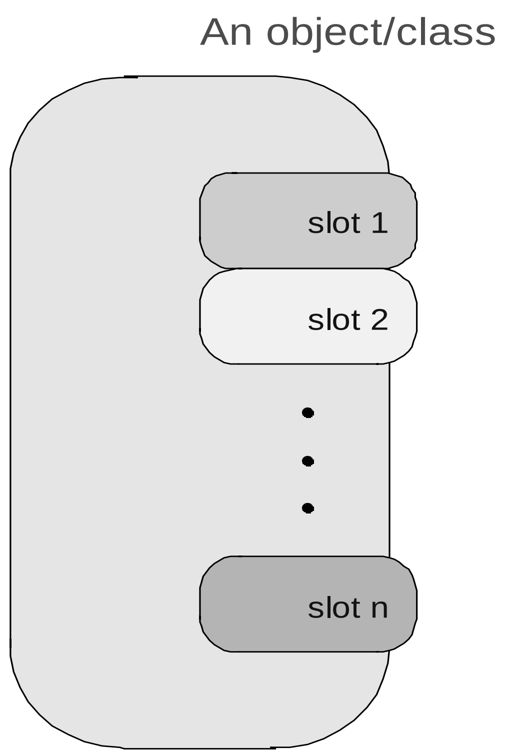
Classes: custom data structure (1/2)

Design your own data class

An object/class



Classes: custom data structure (2/2)



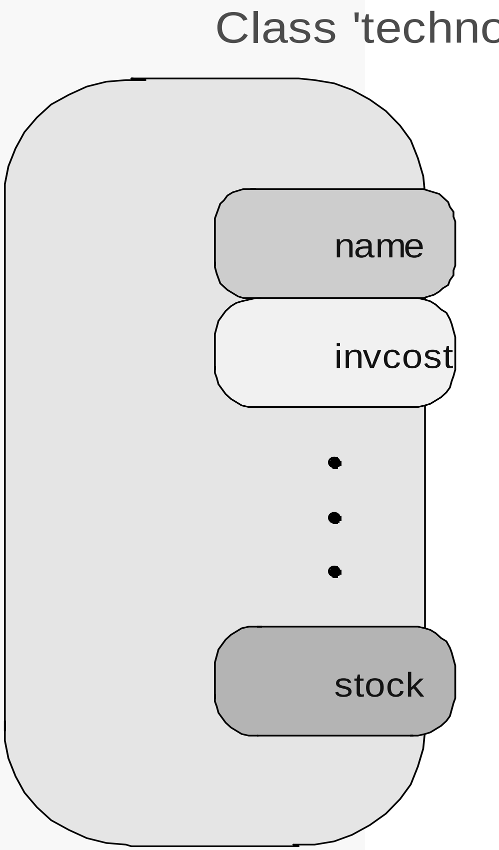
Nestable

energyRt: main classes

- *technology* – stores technology parameters
- *commodity* – stores commodity information
- *supply* – stores info on resources supply
- *demand* – stores info about final demand
- *constrain* – constrain on endogenous variables
- *repository* – stores commodities, technologies, supply, demand
- *model* – stores main model parameters and all info above
- *scenario* – stores all info above and results from the model run

Adding data to “technology” object, final code

```
ECOA <- newTechnology(
  name = "ECOA",
  description = "Coal-fired power plant",
  input = list(comm = "COA",
               unit = "PJ",
               combustion = 1),
  output = list(comm = "ELC",
                unit = "PJ"),
  units = list(capacity = "GW",
               activity = "PJ",
               varom = "MRMB/PJ",
               fixom = "MRMB/GW",
               invcost = "MRMB/GW"),
  cap2act = 31.536,
  afa = list(afa.up = 0.6),
  ceff = list(comm = "COA",
```



energyRt: quick reports for objects *model* and *scenario*

A quick report for scenario ”scen.BAU”

```
report(scen.BAU)
```

September 16, 2016

1

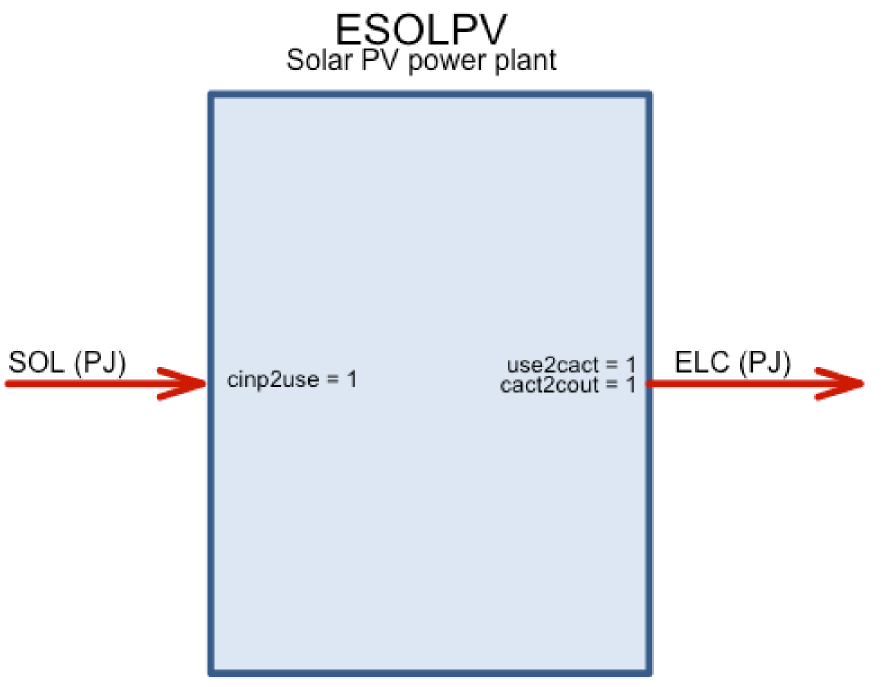
Opt

Co

Quick reports include basic analysis for every commodity, technology, constrain – initial parameters and results

energyRt: current functionality

- Analysis of technologies:
  - Design flexible tech. processes
  - Visualize
  - Analysis of levelized costs
  - Comparison
- Design basic Bottom-Up (RES) models
- Running the models and analyze scenarios



energyRt development plan

- Coming soon
  - Time slices (not yet fully implemented)
  - Regions (not yet tested well)

- Mid-term
  - Check of inconsistencies in data, general bugs
  - Visualizations tools, including:
    - Geographic Information System (GIS)
    - Sankey plots (energy or any commodities flows)
  - Automatic units tracking and conversion
- Long term
  - More models included (Integrated assessment set)

https://olugovoy.github.io/energyRt/

energyRt

energy systems modeling R-toolbox

View on GitHub | Download .zip | Download .tar.gz

Thank you for your attention!

http://energyRt.org

Welcome to energyRt Pages

energyRt is a package for R to develop Reference Energy System (RES) models and analyze energy-technologies.

energyRt package includes a standard RES (or "Bottom-Up") linear, cost-minimizing model, which can be solved by GAMS or GLPK. The model has similarities with TIMES/MARKAL, OSeMOSYS, but has its own specifics, f.i. definition of technologies.