

The MESSAGE_{ix} modeling framework: Building a simple Energy System Model

Volker Krey

Material courtesy of Behnam Zakeri and Daniel Huppmann

NTNU course: Integrated Assessment Modelling (EP8900)

A tutorial to the MESSAGEix framework

Agenda

- Introduction to MESSAGEix framework
- Review of installation
- MESSAGEix model and its structure
- Working with MESSAGEix tutorials (sample models)

Before we start: The MESSAGEix framework installation

1. Install the required software

- Anaconda (Python 3.8, set as PATH environment variable)
- GAMS (64 bit, set as PATH environment variable)



2. Install MESSAGEix

- Open Anaconda Prompt window, and type:
`$ conda config --prepend channels conda-forge`
`$ conda create --name message_env`
`$ conda activate message_env`
`$ conda install message-ix">1"`

3. Download Tutorials (examples for learning the model)

- In Anaconda Prompt window type:
`$messageix-dl --local_path "C:\Users\yourname\myfiles\MESSAGEix"`
(In case of path permission or access error, still Tutorials might have been downloaded)

After this tutorial

The goal is to...

- Learn about the main component of the MESSAGEix framework
- Be able find relevant information about MESSAGEix
- Be able to create a MESSAGEix scenario, solve and plot it using online tutorials
- Learn about some important methods for data handling and scenario analysis

Requirements

- Knowledge on data analysis and programming

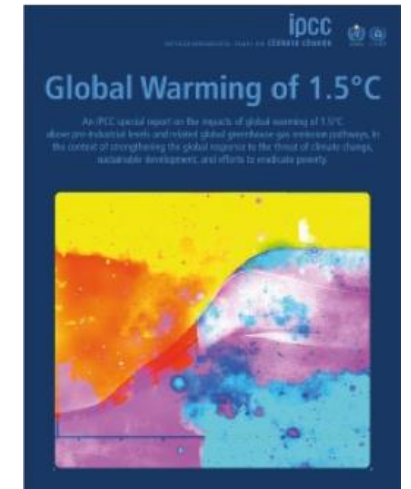
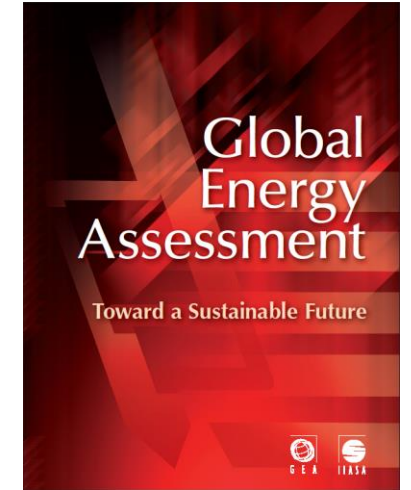


- Knowledge on energy systems
- Patience, motivation, and curiosity

The MESSAGE model at IIASA

IIASA and MESSAGE are at the center of global energy policy assessment

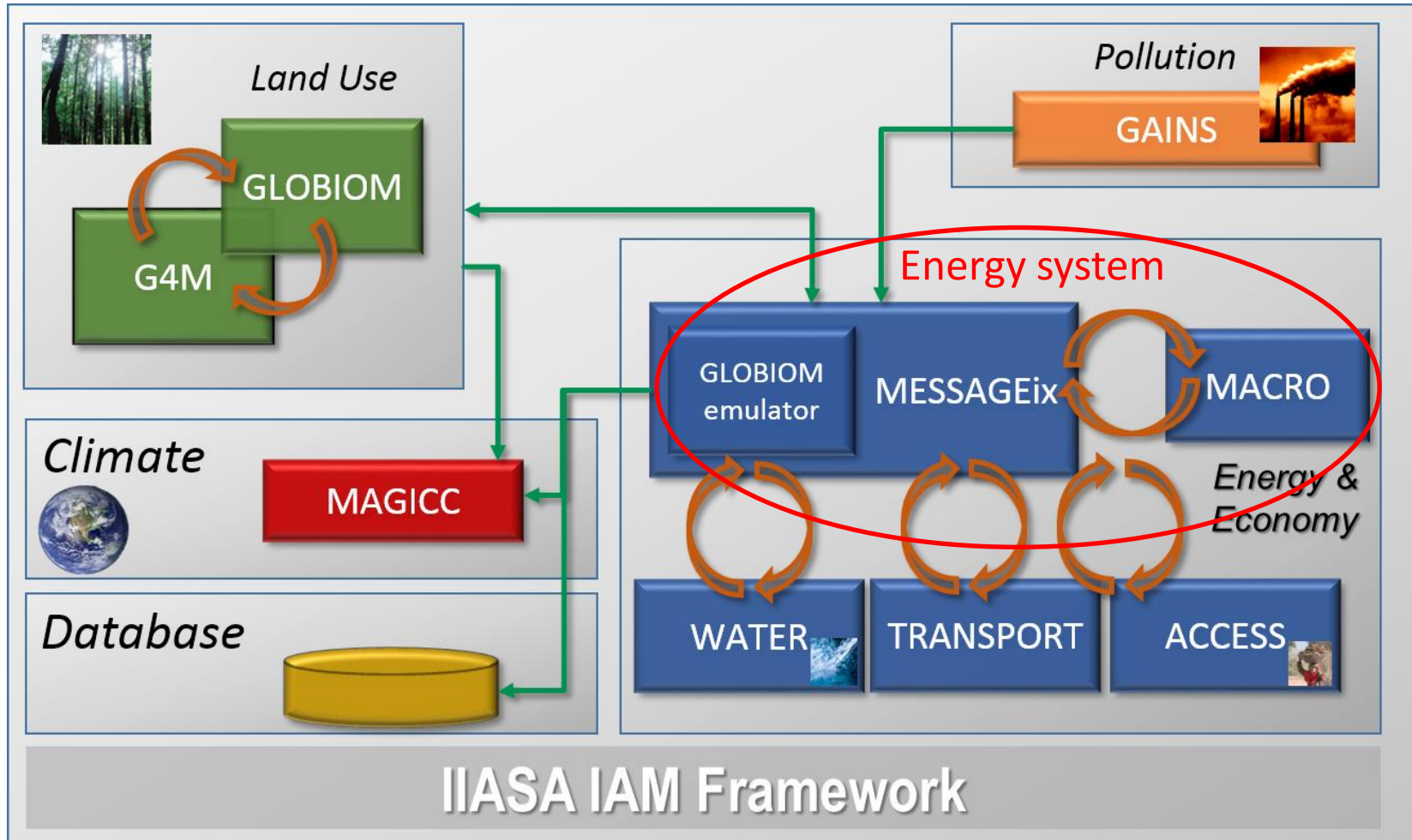
- Since 1980s used for assessing sustainable development and energy/climate policies at regional and global scales
(Manfred Strubegger, Sabine Messner and many others)
- MESSAGE a systems engineering, dynamic optimization model
- MESSAGE widely in use in other organizations like IAEA and member countries



<http://www.ipcc.ch/report/sr15/>

IIASA's Integrated Assessment Model (IAM)

MESSAGEix is at the core



Modeling and Model Development (2)

Discussion in group: what do we expect from a good model?

The MESSAGE_{ix} framework: Goals and Vision

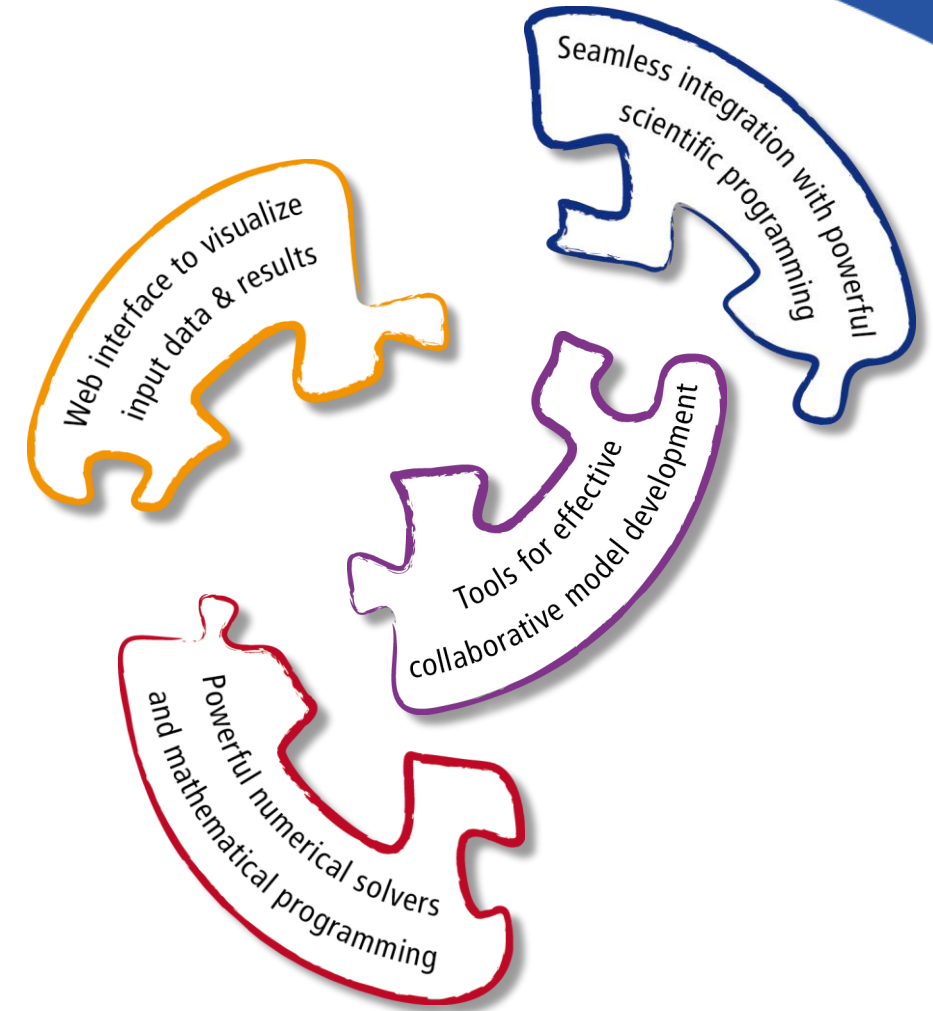
An integrated modeling platform for x-cutting analysis

Goal: Developing a platform for streamlined modeling

- ⇒ using state-of-the-art tools for **data processing**,
- ⇒ building versatile & powerful **mathematical models**,
- ⇒ applying best practice of **collaborative research**

Vision:

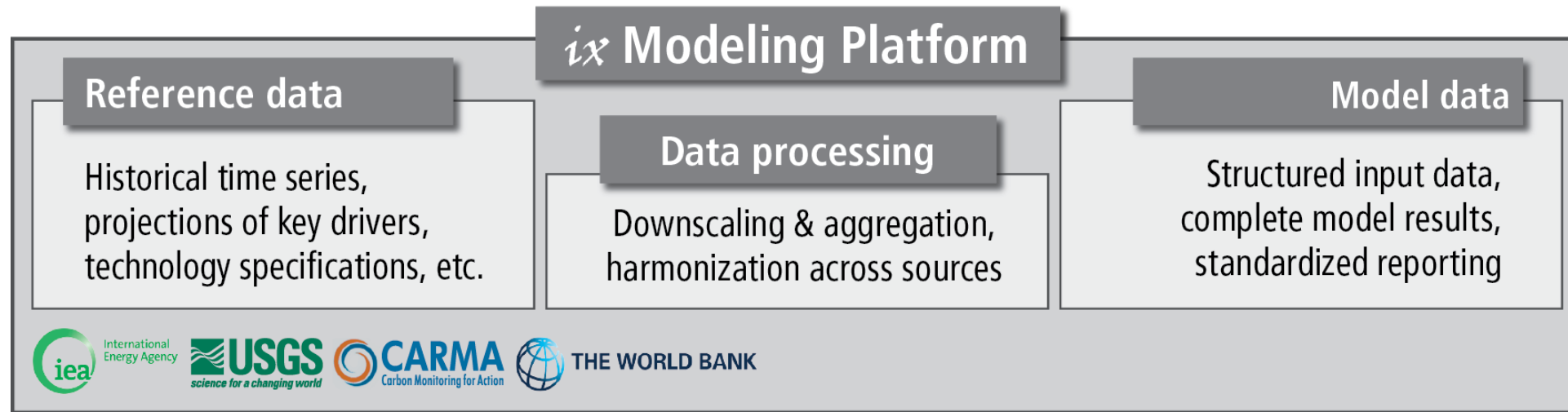
- integration of models & scientific analysis between different disciplines
- across **spatial and temporal** levels of disaggregation
- highest level of transparency and scientific reproducibility for a wide audience



Key features of the *ix* modeling platform

The MESSAGE_{ix} framework: 1. Data management

A central data management system (ixmp)



Good data management is crucial for modeling & scientific analysis:

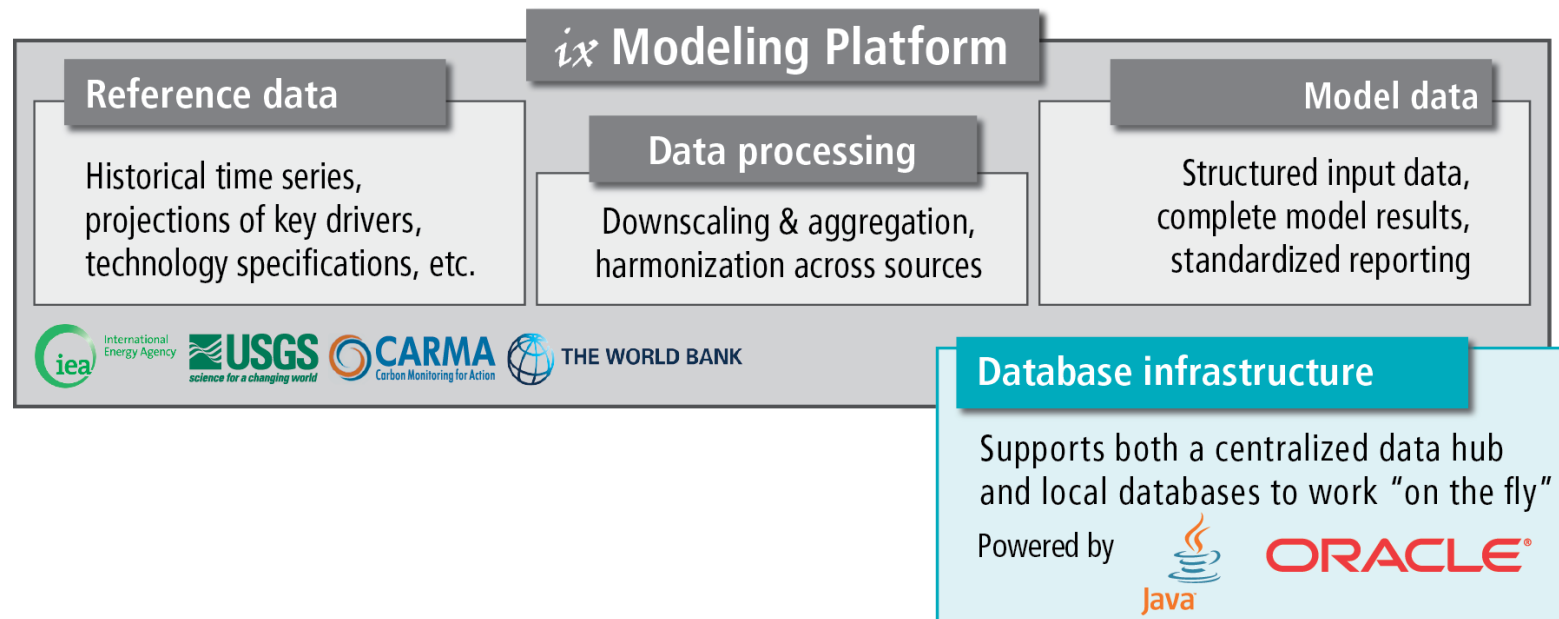
- ... version-controlled and traceable input data for model development
- ... reference data for calibration and verification
- ... efficient workflows based on standardized data processing tools and a common data interface

The MESSAGE_{ix} framework: 2. Database backend

Supported by a high-performance database architecture

The platform...

- ... is based on a Java interface as gateway to the data
- ... supports both an **ORACLE database backend** for high-performance, collaborative modeling and **local, file-based databases** for getting started or working “on the fly”



The MESSAGE_{ix} framework: 3. Integration with GAMS

Connected to high-performance numerical programming

The platform has an interface to GAMS, a versatile software for mathematical programming & optimization.

⇒ MESSAGE_{ix} is the first model fully integrated with the *ix* modeling platform...

Suite of mathematical models

MESSAGE_{ix} & MACRO

Versatile spatial systems-economic model

- ✓ Perfect-foresight or recursive-dynamic approach
- ✓ Easy to add new features & extensions
- ✓ Flexible spatial & temporal detail



G A M S

Water–land integration

The MESSAGE_{ix} framework: Documentation

Implementing tools for comprehensive documentation

The framework ensures transparency and intelligibility through “auto-documentation” of all codes & packages on readthedocs.org

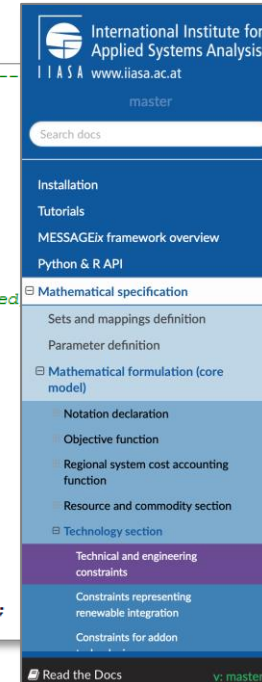
- ⇒ Documentation of all scientific programming packages using ‘sphinx’
- ⇒ Documentation of the mathematical equations generated automatically from $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ mark-up in the GAMS code

 **Read the Docs**

```

***
* Technology section
* -----
*
* Technical and engineering constraints
* ~~~~~
*
* Equation CAPACITY_CONSTRAINT
* ~~~~~
* This constraint ensures that the actual activity of a technology at a node/time cannot exceed available (maintained)
* capacity summed over all vintages, including the technology capacity factor :math:`capacity\_factor_{n,t,y^V,y}`.
*
* .. math::
* \sum_m ACT_{n,t,y^V,y,m,h}
* \leq duration^H_{n,t,y^V,y,h} \cdot capacity\_factor_{n,t,y^V,y,h} \cdot CAP_{n,t,y^V,y}
* \quad t \in T^{INV}
*
* where :math:`T^{INV}` is the set of all technologies
* for which investment decisions and capacity constraints are relevant.
***
CAPACITY_CONSTRAINT(node,inv_tec,vintage,year,time)$( map_tec_time(node,inv_tec,year,time)
AND map_tec_lifetime(node,inv_tec,vintage,year) )..
sum$( mode$( map_tec_act(node,inv_tec,year,mode,time) ), ACT(node,inv_tec,vintage,year,mode,time) )
=L= duration_time(time) * capacity_factor(node,inv_tec,vintage,year,time) * CAP(node,inv_tec,vintage,year) ;

```



Equation STOCKS_BALANCE

This constraint ensures the inter-temporal balance of commodity stocks. The parameter $commodity_stocks_{n,c,l}$ can be used to model exogenous additions to the stock

$$STOCK_{n,c,l,y} + commodity_stock_{n,c,l} = duration_period_y \cdot \sum_h STOCK_CHG_{n,c,l,y,h} + STOCK_{n,c,l,y+1}$$

Technology section

Technical and engineering constraints

The first set of constraints concern technologies that have explicit investment decisions and where installed/maintained capacity is relevant for operational decisions. The set where $T^{INV} \subseteq T$ is the set of all these technologies.

Equation CAPACITY_CONSTRAINT

This constraint ensures that the actual activity of a technology at a node cannot exceed available (maintained) capacity summed over all vintages, including the technology capacity factor $capacity_factor_{n,t,y^V,y}$.

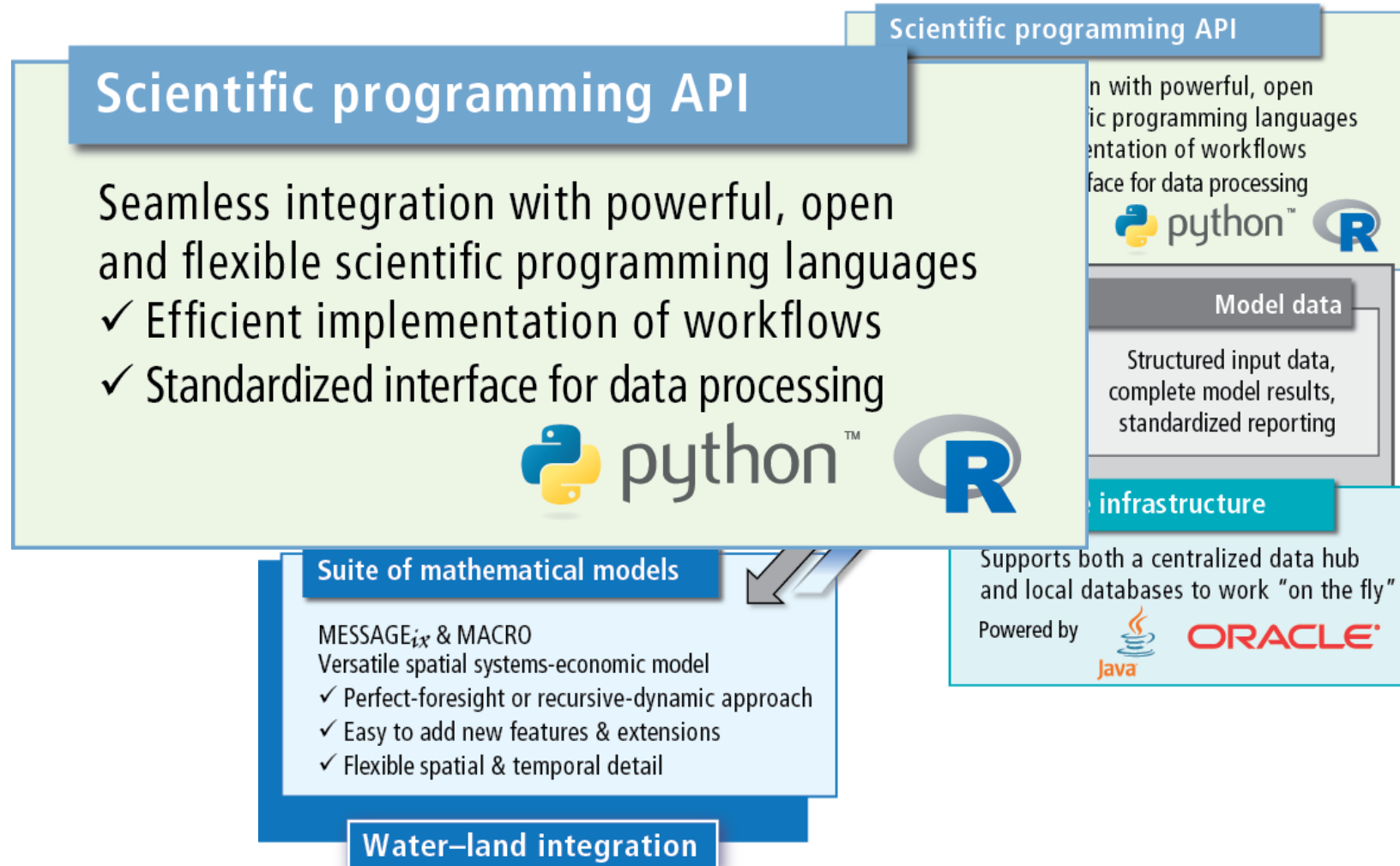
$$\sum_m ACT_{n,t,y^V,y,m,h} \leq duration_time_h \cdot capacity_factor_{n,t,y^V,y,h} \cdot CAP_{n,t,y^V,y} \quad \forall t \in T^{INV}$$

Equation CAPACITY_MAINTENANCE_HIST

The following three constraints implement technology capacity maintenance over time to allow early retirement. The optimization problem determines the optimal timing of retirement, when fixed operation-and-maintenance costs exceed the benefit in the objective function.

The MESSAGE_{ix} framework : 4. Scientific programming

Interfaces to scientific programming for advanced users



The MESSAGEix framework: 5. Collaborative research

Geared towards best-practice in collaborative research

The platform facilitates collaborative model development

... through comprehensive version control of data, model codes and scripts

... implementing “continuous integration”

⇒ automated unit-testing of new features
to ensure stable code base

https://github.com/iiasa/message_ix

- Google Group (questions and answers):

https://groups.google.com/forum/#!forum/message_ix



The MESSAGE_{ix} framework: 6. Interactive web user interface

An intuitive gateway to modeling data for researchers and a wider audience



Web-based user interface

Features

- ✓ Visualization of input data & model results
- ✓ Intuitive drag & drop tables and graphs
- ✓ Data import & export using MS Excel

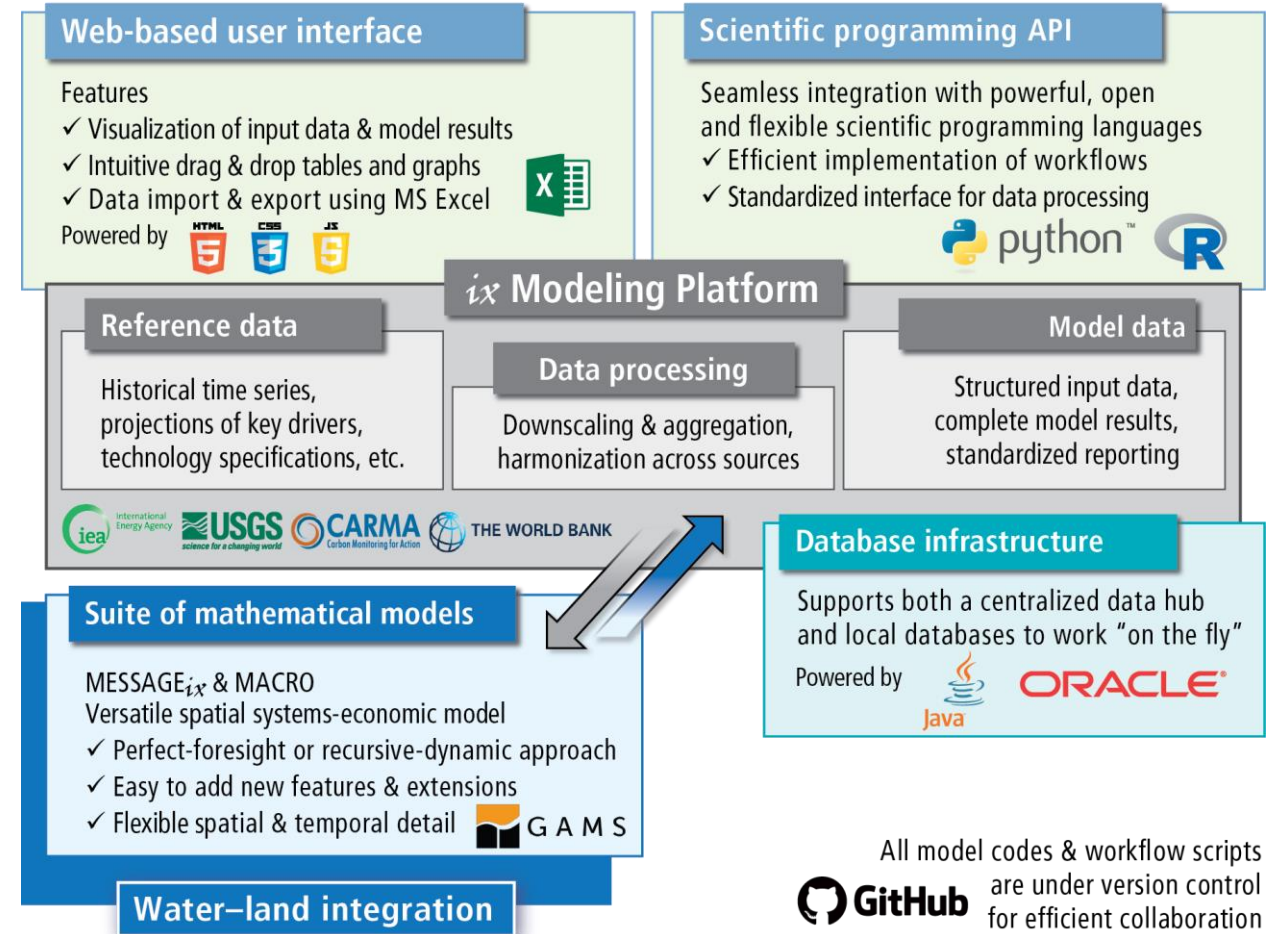
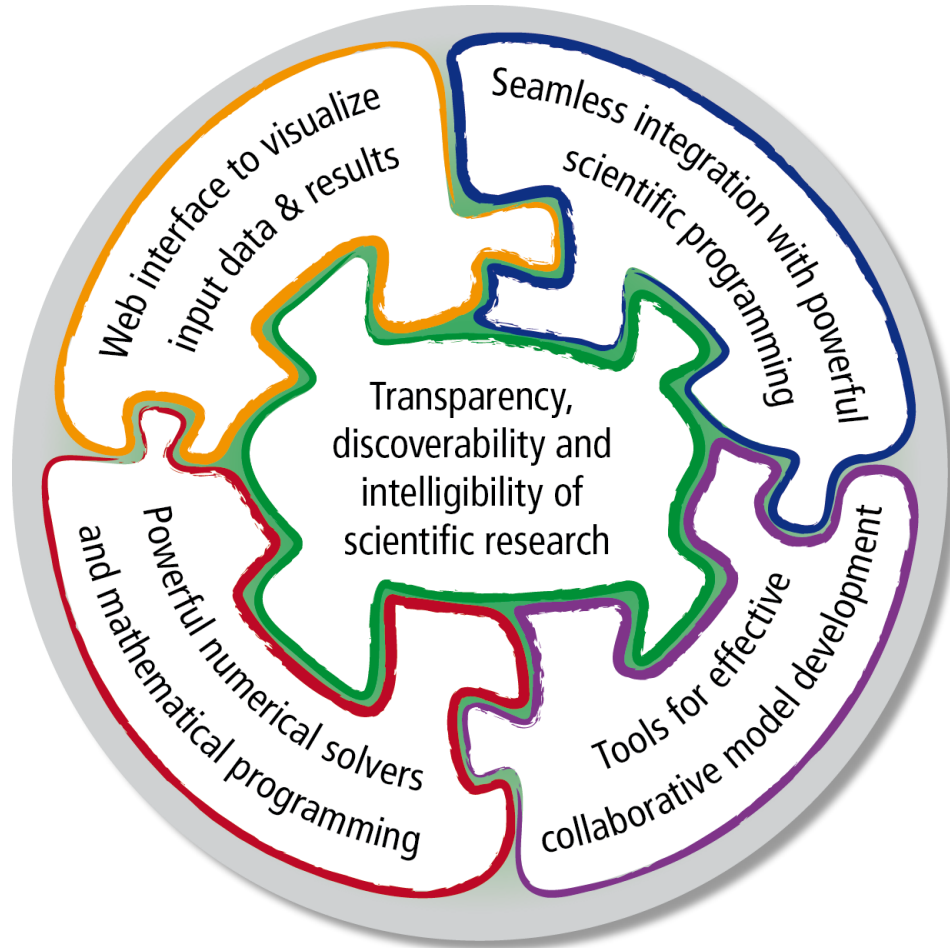


Powered by



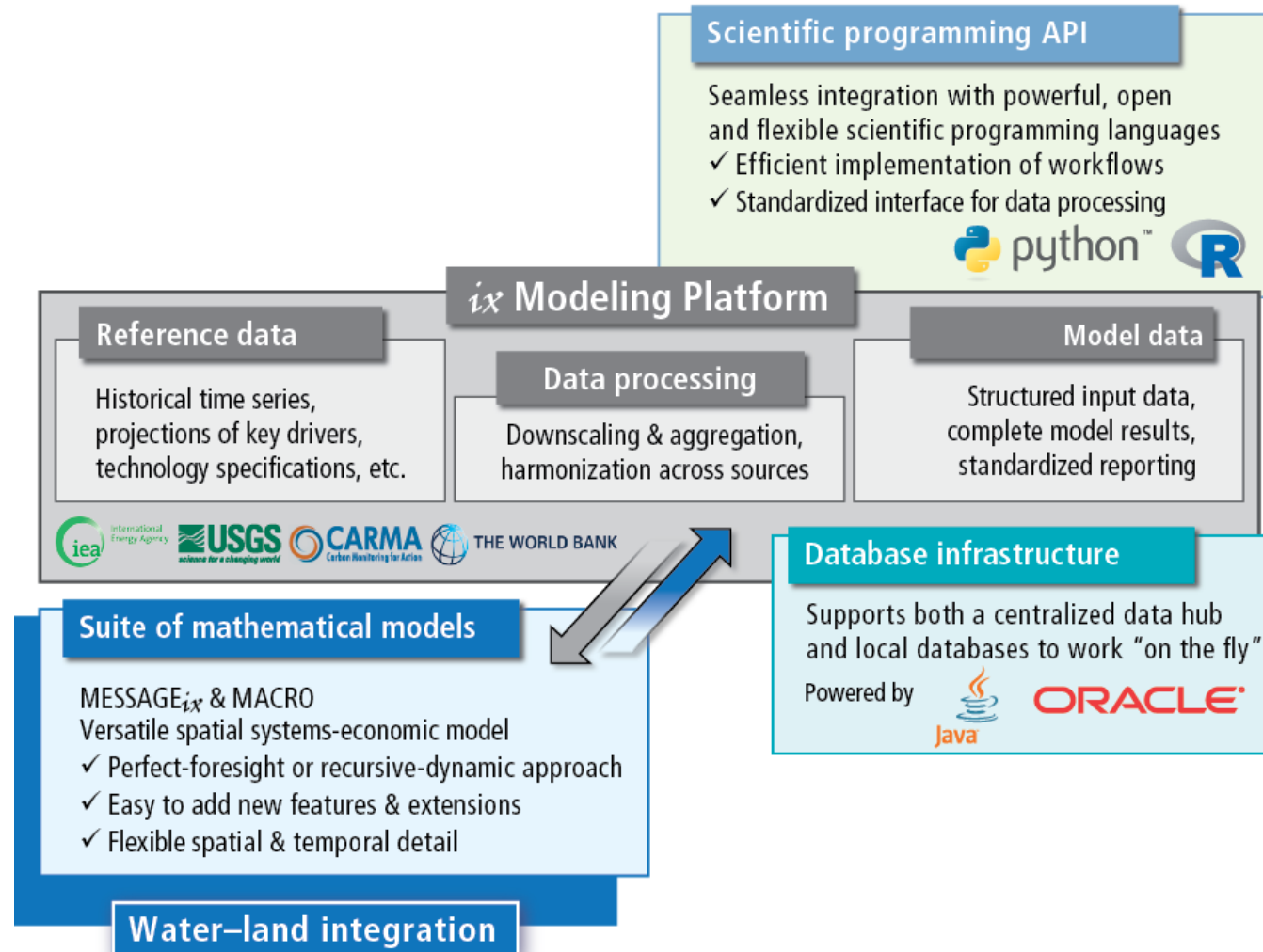
The MESSAGE_{ix} framework

Facilitating transparency and reproducibility of research



The MESSAGE_{ix} framework : Workflow of modeling

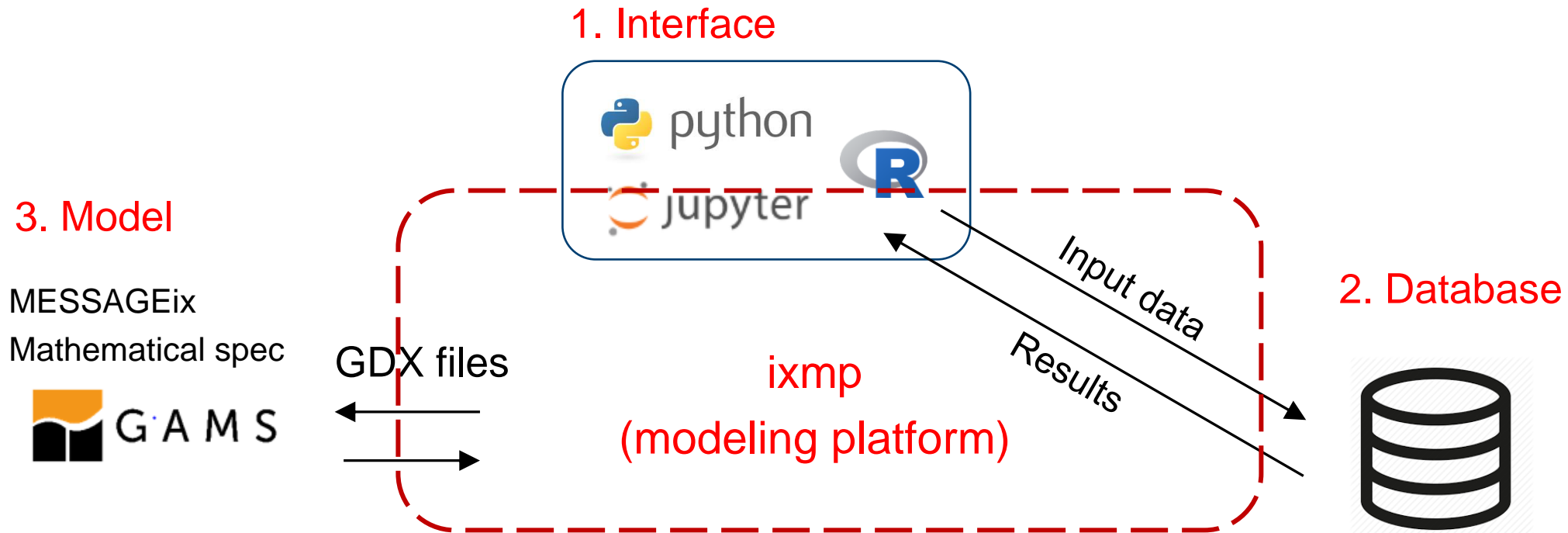
Flexible and high performance processes



The MESSAGE_{ix} framework : Workflow of modeling

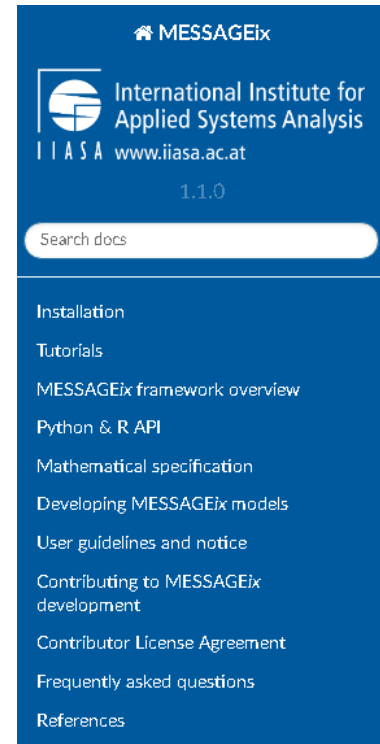
Flexible and high performance processes

- Interface a **central place** for creating, loading, or working with a scenario
- Data can be modified through the interface or other input files (e.g., Excel)
- To see model data and results: load from database, model GDX files, etc.



MESSAGEix Website: main source of information

- Main page:
⇒ <https://docs.messageix.org>
- Open-source GitHub repository:
⇒ https://github.com/iiasa/message_ix
(contribution guide)
- Tutorials and examples



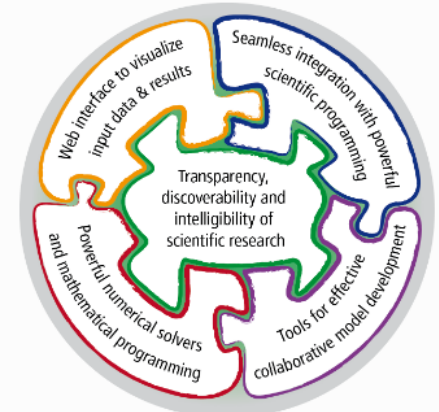
[Docs](#) » The MESSAGEix framework

[View page source](#)

The MESSAGEix framework

Overview and scope

MESSAGEix is a versatile, open-source, dynamic systems-optimization modelling framework. It was developed for strategic energy planning and integrated assessment of energy-engineering-economy-environment systems (E4). The framework can be applied to analyse scenarios of the energy system transformation under technical-engineering constraints and political-societal considerations. The optimization model can be linked to the general-economy MACRO model to incorporate feedback between prices and demand levels for energy and commodities. The equations are implemented in the mathematical programming system **GAMS** for numerical solution of a model instance.



The ix modeling platform (source: [1])

The MESSAGEix framework: Installation

Two types of installation

1. Install MESSAGEix through Anaconda

- You only want to use the public release (latest version)
- You don't aim to contribute to the code on Github
- You will be still able to develop your features to your code

2. Install through the source (advanced users, developers)

- You want to test and review the latest features (i.e., the ones after the latest public release)
- You want to contribute to the code
- You know about the collaborative code development

The MESSAGEix framework: After installation

What is where?

1. ixmp and message_ix python packages

- Locate your Anaconda (python) library and navigate to *site-packages*
 - You can open windows command line and type:
`$where Anaconda`
 - Then, navigate to *site-packages* (C:\...\Anaconda3\Lib\site-packages)
- You should be able to see two packages *ixmp* and *message_ix*

2. Mathematical model folder

- Model folder is by default under message_ix folder
- Model folder can be changed to a new folder (optional)

Open Conda command window, and type:

```
$messageix-config --model_path /path/to/model
```

The MESSAGEix framework: Mathematical formulation

Everything is at your disposal!

1. Looking into the GAMS files

- Locate your “model” folder (for example, C:\...\message_ix\model)
- Create a GAMS project there (optional)
- MESSAGE formulation in (C:\...\model\MESSAGE): for example look at **model_core.gms**
- Input data in (C:\...\model\data): GDX files
- Output results in (C:\...\model\output): GDX files

Linear programming (LP) (reminder)

Applications of LP

- Production management
- Personnel management
- Marketing management
- Resource/ inventory management
- Blending problem, etc

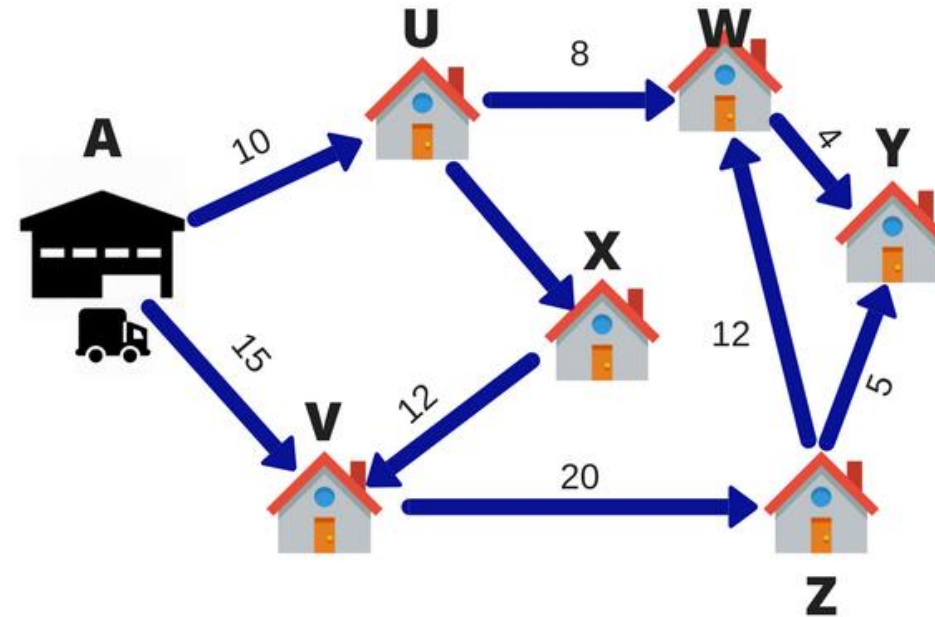


image: deepai.org

Principle:

Maximizing the utility (use of resources)

Or Minimizing the cost of meeting a service → MESSAGEix

MESSAGEix: a linear optimization model

Minimizing cost of the system

- **Objective:** The least cost option for meeting certain services (demand)
- **System:** a set of technologies (processes), resources, and products
- **Cost of the system:** installing/maintaining **capacity**, cost of **activity** (operation), and emissions, land use and relation costs (if any)
- **Constraints:** maximum use of a technology, growth/decline rates of activity, capacity factor, etc.

A note on “capacity” and “activity” (MESSAGEix formulation)

- **Capacity:** installed units of a technology (e.g. three cars)
- **Activity:** operation of that technology (e.g., 3 h usage of each car per day)
- **Reminder:** capacity \neq activity

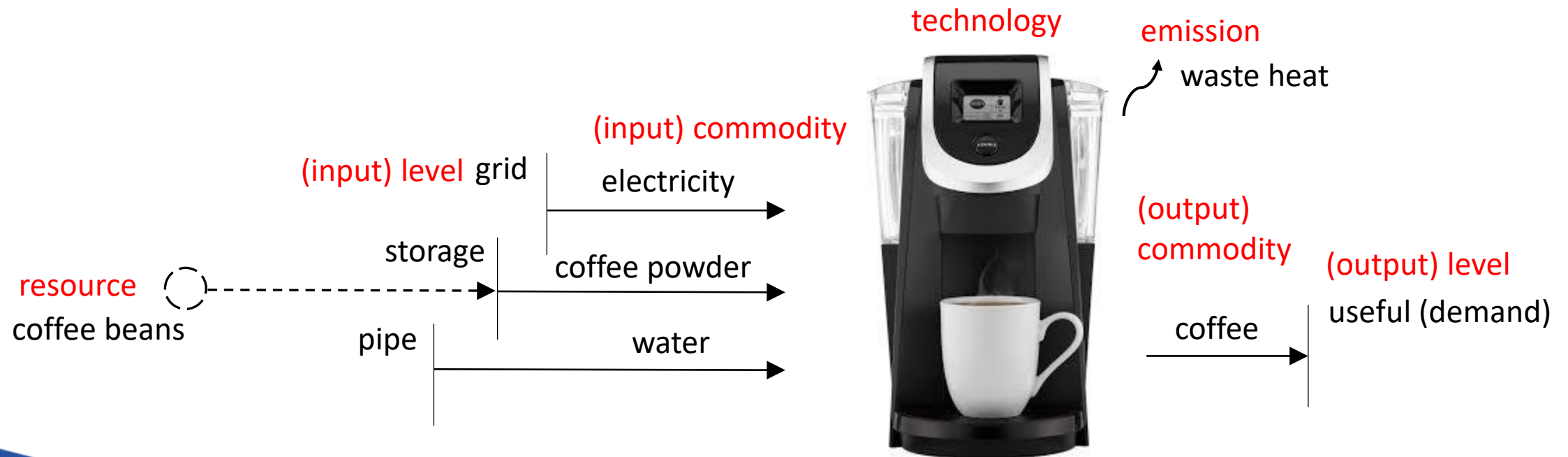


MESSAGEix: a technology-based model

Technologies and resources meet demand

- Example **technologies**: electric car, reactor, pipeline, power plant, building, ship, industrial process

A sample technology: coffee machine



MESSAGEix: Sets and Parameters

Sets:

- List of elements for building a model
- Example: technology, commodity, level, node, etc.

CONSTRUCTION MATERIALS		
Part	Cast Iron Pump	Stainless Steel Pump
Inlet / Motor Bracket	Cast Iron	Cast 316 SS
Discharge Housing	Steel	316 SS
Pump Housing	304 SS Tubing	316 SS Tubing
Impellers	Noryl	Noryl
Diffusers	Noryl	Noryl
Wear Rings	316 SS	316 SS
Shaft & Coupling	316 SS / 416 SS	316 SS
Shaft Bearing Sleeve	316 SS	316 SS
Shaft Bearing	Bronze	Rulon
Mechanical Seal	Carbon/Ceramic	Carbon/Ceramic
Mechanical Seal Spacer	302 SS, Buna N	316 SS, Viton
O-Rings	416 SS	316 SS
	Buna-N	Viton

Parameters:

- Adding information about sets
- Defining quantities (specification)
- Relating sets to each other, etc.



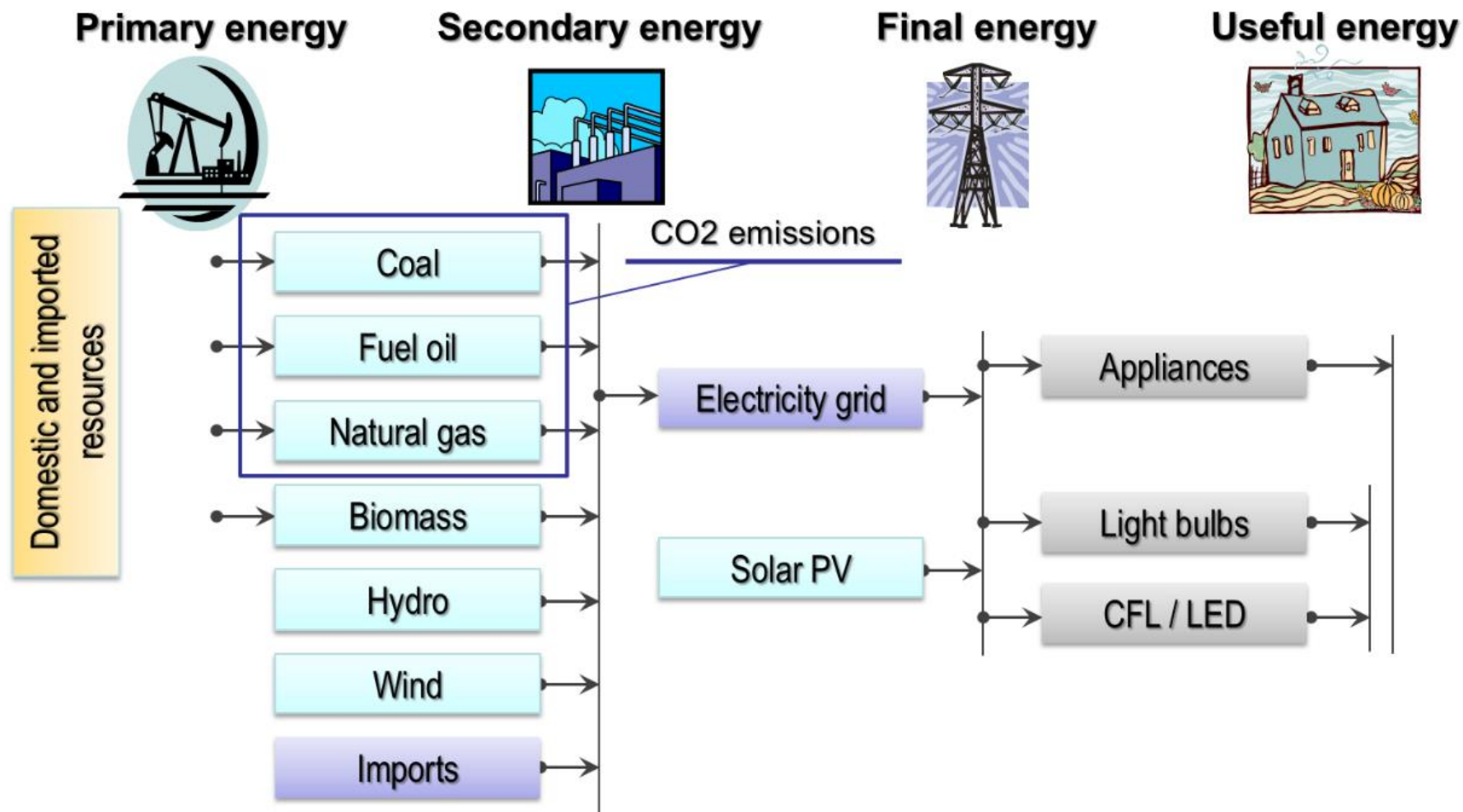
Equations:

- Relationship between sets, parameters, **variables**, etc.
- Building the model



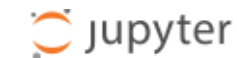
MESSAGEix: Energy system

Technologies, commodities, levels, etc



- Locate your tutorial folder in your machine
- Then, open a command window and call *jupyter notebook*
- Navigate to the folder for Westeros tutorials and open the baseline

This tutorial is based on the country of Westeros from the TV show "Game of Thrones".



Select items to perform actions on them.

30

Westeros baseline

Different steps of modeling

- Creating a new scenario (or loading an existing one)
- Declaring required sets (*node, technology, commodity, level*, etc.)
- Defining required parameters (adding numeric data, relating sets to each other, etc.)
 - *demand*
 - *techno-economic parameters*
 - *bounds and dynamic constraints*
- Solving the model
- Postprocessing and plotting

Working with MESSAGEix scenarios

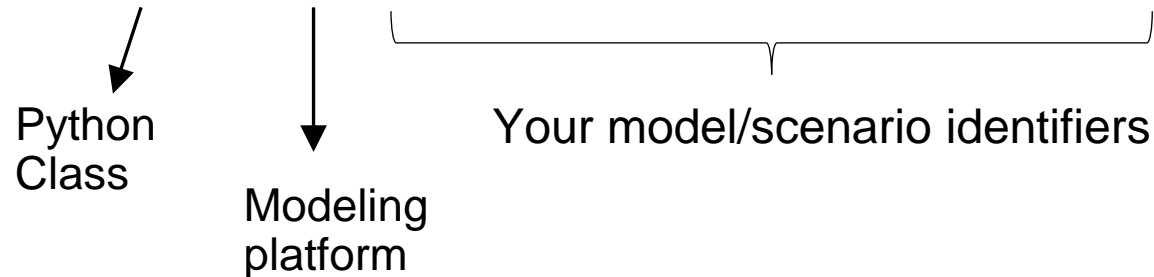
A short note on model/scenarios

- Loading the ixmp platform (connection to the database):

```
mp = ixmp.Platform(dbtype='HSQLDB')
```

- Creating a new scenario:

```
my_scen = message_ix.Scenario(mp, model, scenario, version='new')
```



Example: `model = 'building energy system', scenario = 'baseline' (or 'low efficiency')`

- Loading an existing scenario:

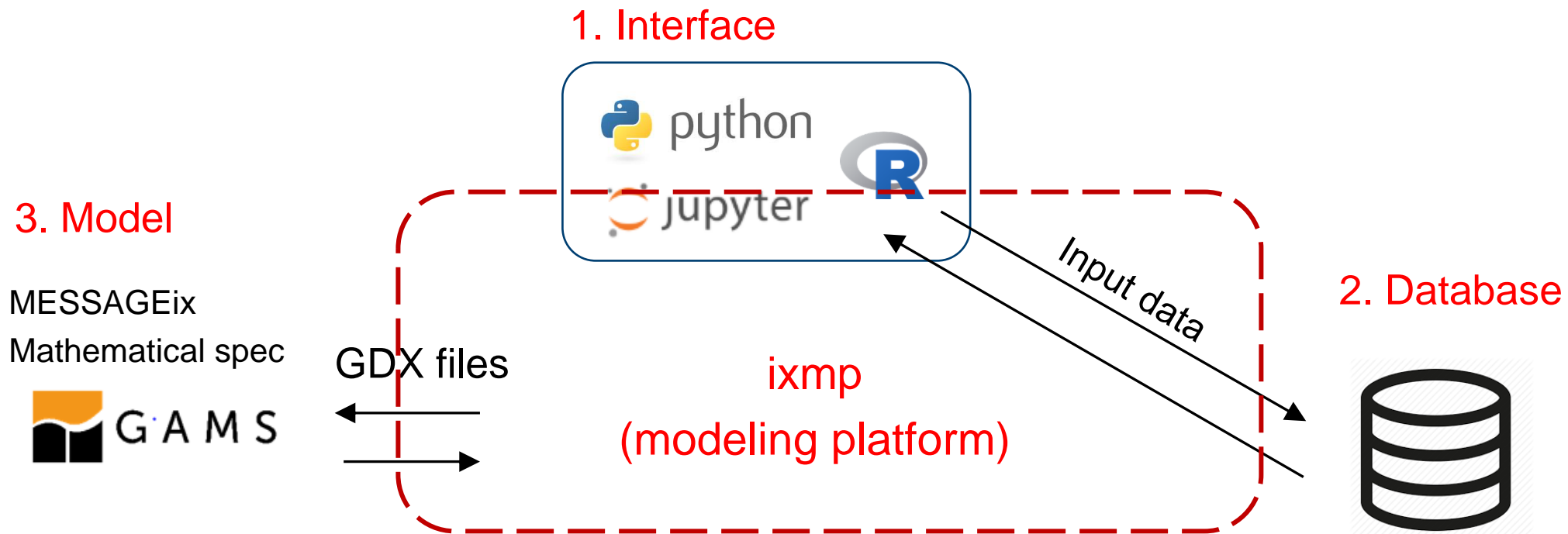
```
my_scen = message_ix.Scenario(mp, model, scenario, version=1)
```

```
my_scen = message_ix.Scenario(mp, model, scenario)
```

The MESSAGE_{ix} framework : Workflow of modeling

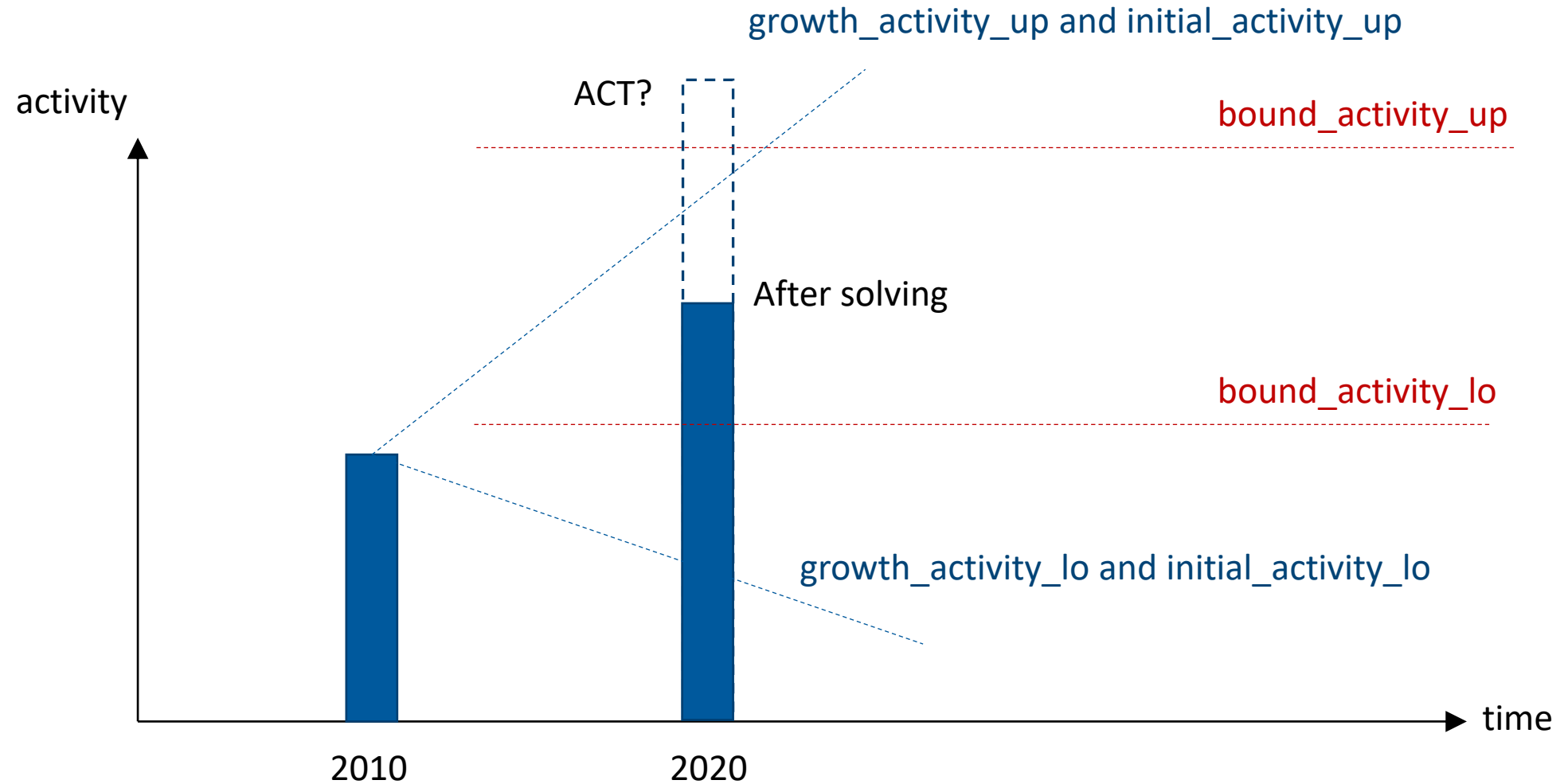
Reminder

- Interface a **central place** for creating, loading, or working with a scenario
- Data can be modified through the interface or other input files (e.g., Excel)
- Model data and results: loaded from database, model GDX files, etc.



Dynamic constraints

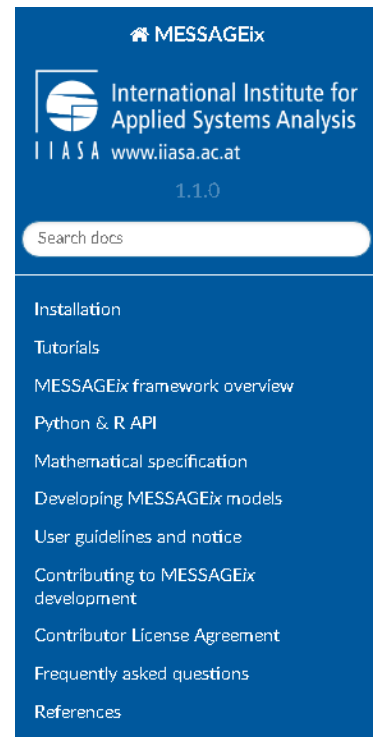
Controlling behavior of technologies over time



[Link to the documentation](#)

More information

- Main page:
⇒ <https://docs.messageix.org>
- Open-source GitHub repository:
⇒ https://github.com/iiasa/message_ix
(contribution guide)
- Tutorials and examples
⇒ https://teaching.ece.iiasa.ac.at/message_ix_workshop.html



[Docs](#) » The MESSAGEix framework

[View page source](#)

The MESSAGEix framework

Overview and scope

MESSAGEix is a versatile, open-source, dynamic systems-optimization modelling framework. It was developed for strategic energy planning and integrated assessment of energy-engineering-economy-environment systems (E4). The framework can be applied to analyse scenarios of the energy system transformation under technical-engineering constraints and political-societal considerations. The optimization model can be linked to the general-economy MACRO model to incorporate feedback between prices and demand levels for energy and commodities. The equations are implemented in the mathematical programming system [GAMS](#) for numerical solution of a model instance.



The ix modeling platform (source: [1])

- Scientific reference to the MESSAGE_{ix} framework:
Huppmann et al. (2019). *The MESSAGEix Integrated Assessment Model and the ix modeling platform: An open framework for integrated and cross-cutting analysis of energy, climate, the environment, and sustainable development*. Environmental Modelling & Software 112: 143-156. DOI: [10.1016/j.envsoft.2018.11.012](https://doi.org/10.1016/j.envsoft.2018.11.012)

Thank you very much for your attention!

Volker Krey

Integrated Assessment & Climate Change (IACC) Group

Energy, Climate & Environment (ECE) Program

International Institute for Applied Systems Analysis (IIASA)

Laxenburg, Austria

krey@iiasa.ac.at

www.iiasa.ac.at