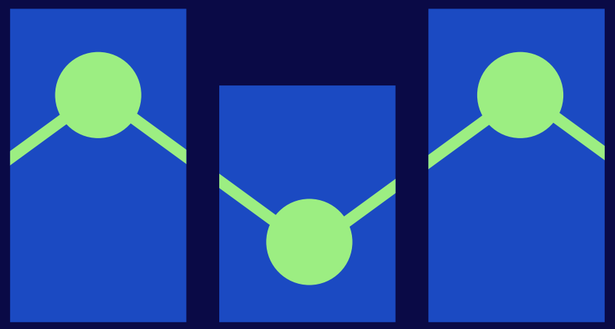
mopo



Spine Toolbox

Open-source workflow tool for flexible scenario and data management

Funded by
the European Union

TABLE OF CONTENT

ACRONYMS

API	Application Programming Interface
BRELL	Belarus Russia Estonia Latvia Lithuania
DAG	Directed Acyclic Graph
GIS	Geographical Information System
GUI	Graphical User Interface
INES	Interoperable Energy System
SQL	Structured Query Language

Spine Toolbox provides a **workflow, data and scenario management framework** that can combine multiple data sources and tools while giving the user a full view of the workflow from sources to outcomes. It allows **groups of users to collaborate on large-scale problems** that require data curation as well as multiple tools and models. Toolbox and the data structure features are capable of supporting a wide range of **data processing and modelling tasks** with a focus on data consolidation and scenario analysis.

INTRODUCTION

SPINE TOOLBOX TECHNICAL DESCRIPTION

Spine Toolbox is an **open-source graphical user interface** to manage data, scenarios and workflows for modelling and simulation. Users can have local workflows, but work as a team through version control and SQL databases.
Its architecture combines **workflow orchestration with structured data and scenario management** in a single modular environment.
Workflows are represented as **DAGs**, where each computational node receives input from predecessor nodes, performs internal operations, interacts with external data sources and pushes outputs downstream. This structure enables **transparent dependency handling and reproducible execution.**

The **data layer** is based on a flexible Entity-Attribute-Value model with classes and relationships, allowing problem-independent data representation and systematic scenario construction through alternative parameter values.
Execution is handled by Spine Engine, which constructs pipelines from workflow items and enables parallel processing of independent nodes and scenarios. The Engine supports both GUI-based and headless execution modes, as well as client-server deployment for remote computation, ensuring scaleability and efficient use of multi-core hardware.

FEATURES

INTRODUCTION

SPINE TOOLBOX FEATURES

DATA MODEL AND MANAGEMENT

FLEXIBLE OBJECT-ORIENTED DATA MODEL

The Spine Data Structure applies a generic Entity-Attribute-Value approach with classes and relationships, implemented as SQL databases via SQLAlchemy. Its abstract structure supports **diverse modelling needs in a scaleable, object-oriented way**.

ADVANCED PARAMETER HANDLING

Entities can store constants, time series, arrays and multi-dimensional maps. The interface enables intuitive browsing, editing and structured management of **complex parameter sets**.

METADATA

Entities and parameters in Spine Data Structure can be associated with **arbitrary metadata**. This allows the data and its provenance to be described in detail.

SUPPORT FOR GIS DATA AND GEOJSON

Entities in the Spine Data Structure can include geographic coordinates and geometries encoded in GeoJSON format, enabling spatial representation and geospatial analysis.

DATA INTEROPERABILITY AND DATA EXCHANGE

DATA INTEROPERABILITY

Importer and exporter tools allow conversion across Spine Data Structure and popular formats such as CSV, Microsoft Excel (.xlsx), GAMS Data Exchange (.gdx) and SQL databases. This interoperability enables **integration with external tools** and supports a **wide spectrum of analytical workflows**.

APACHE ARROW SUPPORT

The Python API for Spine Data Structures can readily interoperate with Apache Arrow data format, which in turn works as a **data exchange format** for popular data analytics frameworks such as Pandas or Xarray.

WORKFLOW EXECUTION AND ORCHESTRATION

TOOL SPECIFICATION FRAMEWORK

Tool specifications formally define execution requirements, inputs, outputs and dependencies for each workflow component. This ensures structured **orchestration, transparency and reproducibility across modelling pipelines**.

PARALLEL AND REMOTE EXECUTION ENGINE

Execution is separated from the interface, enabling parallel processing of workflow nodes and scenarios. The **built-in client-server system supports remote execution, improving** scalability and efficient use of computational resources.

LOOPS

Spine Toolbox workflows need not be strictly acyclic, even though DAG stands for directed acyclic graph. Toolbox supports **conditional backward jumps**, which allow looping over segments of a workflow. This enables, for example, rolling or stepwise execution of tools and iterative approaches.

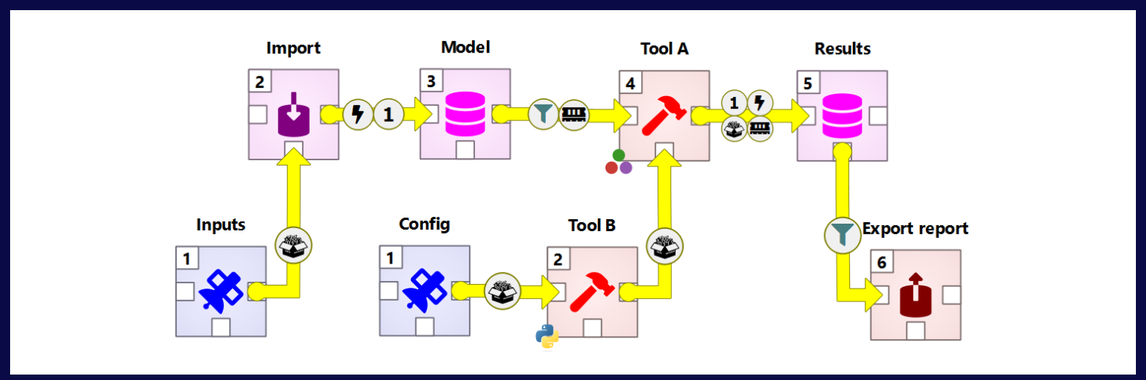


Figure 1. Example of a simple Spine Toolbox workflow showing tool dependencies and execution sequence.

MODELLING LOGIC AND SIMULATION CAPABILITIES

BUILT-IN SCENARIO ENGINE

Scenarios are constructed from **alternative parameter values within the same dataset**. Dedicated interfaces enable **systematic comparison** of model configurations and uncertainty analysis.

TRANSPARENT DATA TRANSFORMATIONS

Workflows can incorporate data transformers that modify Spine Data Structure data transparently **without touching the source data**.

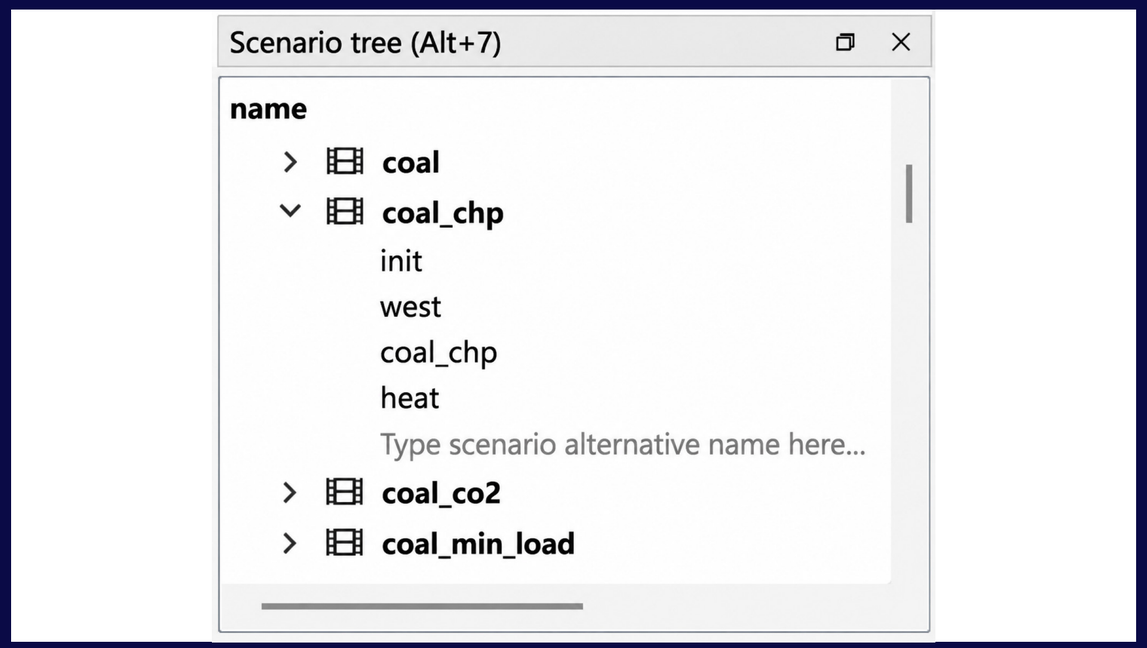


Figure 2. Scenario tree illustrating the combination of parameter alternatives used to generate and compare model scenarios.

EXTENSIBILITY AND INTEGRATION ECOSYSTEM

DUAL-LANGUAGE API ACCESS

Python-based Spine Database API enables **advanced querying and automation** of Spine datasets. A parallel Julia interface supports extended capabilities, including direct model construction and integration with optimisation frameworks.

EXTENSIBLE PLUG-IN ARCHITECTURE

Tool specifications can be **converted into plug-ins**. An example is the SpineOptPlugin which provides readily configured tools for the SpineOpt energy system model. Plugins are installed as toolbars in the Toolbox GUI from where they can be dragged and dropped to the workflow.

```python
url = "sqlite:///home/spineuser/data/fish.sqlite"

with DatabaseMapping(url) as db_map:
    db_map.add_entity_class(
        dimension_name_list=("fish", "cat"),
        description="A fish getting eaten by a cat?",
    )
    db_map.commit_session("Added a relationship.")
```

Figure 3. Example of Python code using Spine Database API for data access, querying, and workflow automation.

USER INTERACTION, VISUALISATION AND COLLABORATION

DATA VISUALISATION AND EDITING

Spine Toolbox includes Spine Database Editor, which allows different views into Spine Data Structures. The **tabular view** presents data in spreadsheet-like tables for direct access and editing, while the **pivot view** enables custom table construction through drag-and-drop of rows and columns. The **graph view** offers a visual representation of entities and their relationships, with optional integration of GIS data when available.

VERSION-CONTROLLED WORKFLOW SHARING

Spine Toolbox provides features that facilitate the **sharing of workflows** via version control systems such as Git. A dedicated mode is available for users working with workflows authored by others, allowing them to modify the workflow **without creating conflicts with the original version** provided by the author.

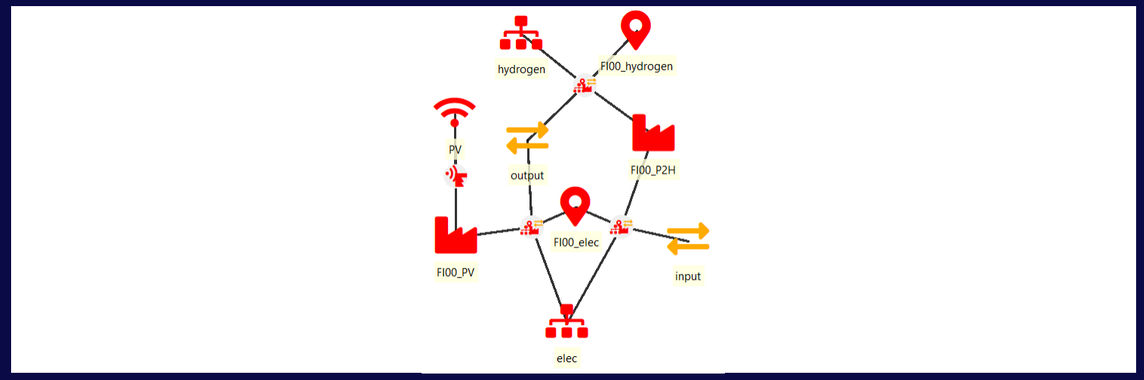


Figure 4. Graph View of Spine Database Editor, visualising entities and their relationships within a Spine data structure.

USE CASES

USE CASE 1

PAN-EUROPEAN ENERGY SYSTEM MODEL

The Pan-European energy system is transforming to reach **climate neutrality** by 2050, guided by the European Green Deal targets: -55% GHG by 2030, -90% by 2040, and net zero by 2050. This requires **coordinated action and robust modelling**. A **unified framework** integrates open data into sectoral models, adaptable by detail, geography, covering renewables, networks, storage, end-use. To know about this use case and how the Spine Toolbox features are used, visit tools-for-energy-system-modelling.org/use-cases/pan-european-energy-system-model/.

FLEXIBLE OBJECT-ORIENTED DATA MODEL

EXPLICIT INTERMEDIATE DATABASES

The workflow for the Pan-European dataset uses **intermediate databases between the raw data and the final-model database**. These intermediate databases reshape the raw data and convert it into a **structured format of entities and their relationships**—for example, generation or storage technologies, their connections to nodes, the nodes themselves or the flows between nodes. In addition, they enable **more effective mapping to the final-model database** by linking source and target entities.

ADVANCED PARAMETER HANDLING

SECTORAL DATA PIPELINES

The case study connects **multiple pipelines** from different sectors of the energy economy, and the dataset contains a wide variety of parameter types: time series for capacity factors, demand and inflows; constants used to define costs or technical parameters; and mapping structures that link cost parameters to optimisation periods. The intermediate databases enable **inspection of these data and support improved data curation**.

BUILT-IN SCENARIO ENGINE

INTUITIVE SCENARIO BUILDING

The European energy-system database contains data with many possible **configuration alternatives**. Thanks to Spine Database API and Spine Database Editor, building the scenarios used for long-term system planning under future climate policies becomes highly intuitive and straightforward. This framework enables efficient construction and analysis of a wide range of scenarios with ease.

DATA INTEROPERABILITY

INTEROPERABILITY WITH VARIOUS TOOLS

The framework developed for the European case study incorporates several **interoperability features** enabled by Spine Toolbox. First, much of the raw data

is provided in YAML, CSV, and Excel formats, but thanks to the data-connection mechanisms, importing and especially updating information within the workflow is straightforward. Second, Spine Toolbox supports the **construction of frameworks with interoperable databases**. For example, the model is built using an interoperable energy specification that can be linked, through transformers, to different energy-system optimisation tools.

DUAL-LANGUAGE API ACCESS

MODEL BUILDER

The European case study includes data pipelines structured into three components: data connectors, importers and SQLite databases. These components are linked to a **final Spine database through a model builder**. Both the model builder and the importers are Python scripts that use Spine Database API to efficiently read from and write to the databases, thereby **streamlining and automating data flows throughout the workflow**.

TOOL SPECIFICATION FRAMEWORK

MODEL CONFIGURATION

The framework for both the importers and the model builder is implemented using Python-based tools. For the data pipelines, the inputs consist of raw data, the template structure of the databases, and configuration files, while the output is the generation of the corresponding database for each pipeline. For the model builder, the **inputs are the pipeline databases and configuration files**, and the output is the **energy-system model expressed in the INES format** (Interoperable Energy System Data Specification).

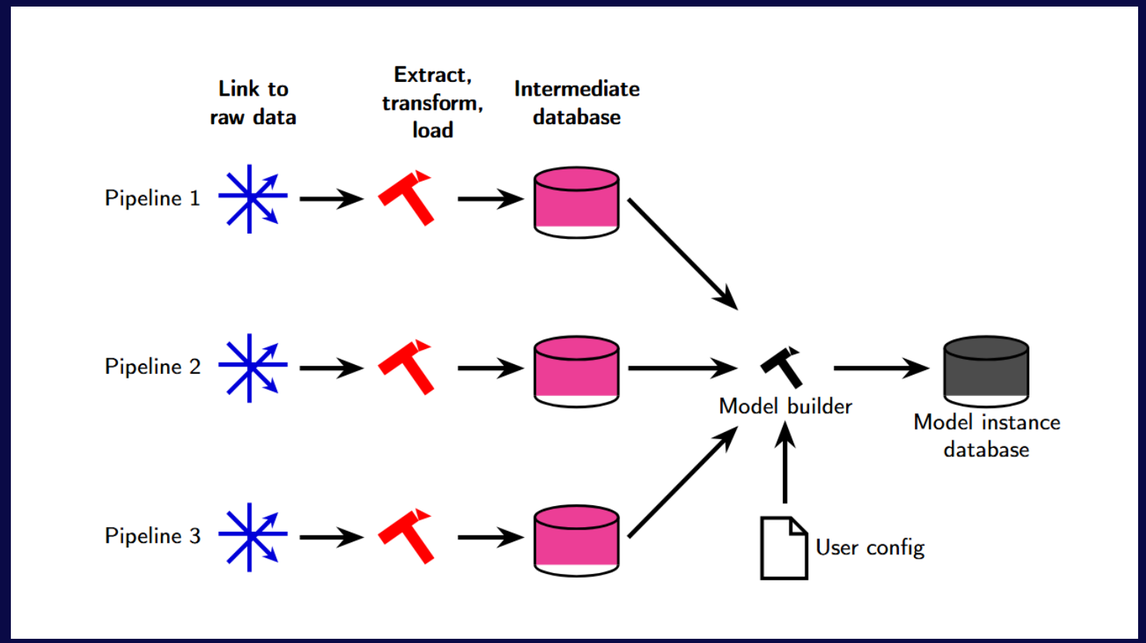


Figure 5. Spine Toolbox workflow for the Pan-European case study, integrating multiple data pipelines into a single model instance database.

USE CASE 2

BALTICRESTORE

Since the disconnection from BRELL in 2025, the Baltic energy system faces **challenges from rising renewable variability**. In Mopo, the Baltic case study addresses large-scale wind integration under spatial constraints using high-resolution data. The **Baltic model built in the Backbone energy system model format is translated via Spine Toolbox** to INES and SpineOpt, demonstrating the modelling workflow, interoperability and analytical capabilities. To know about this use case and how the Spine Toolbox features are used, visit tools-for-energy-system-modelling.org/use-cases/balticrestore/.

FLEXIBLE OBJECT-ORIENTED DATA MODEL

MODEL-TO-MODEL DATA CONVERSION VIA INES

Throughout the Baltic case study workflow, the model is represented at each intermediate stage as an SQL database in the Spine format, but adhering to different specifications. These stages include the **Backbone model imported into a Spine database** (*bb_db_model*), the **model following the INES specification** (*ines_db_model*), and finally the **SpineOpt database** (*spine_db_model*).

ADVANCED PARAMETER HANDLING

DIVERSE PARAMETER TYPES

In each of the databases comprising the overall workflow, the Baltic model is expressed through **zero and multi-dimensional entities with parameter-value pairs.** The values can take a wide range of forms enabled by the Spine Data Structure, Spine Database Editor and the respective specification, including constants, time series, maps etc.

BUILT-IN SCENARIO ENGINE

SCENARIO SELECTION

When executing the SpineOpt model at the end of the Baltic case study workflow, it is possible to select from a number of scenarios, defined through the **combinations of alternatives for the database parameter values.**

DATA INTEROPERABILITY

MODEL SHARING AND INTEROPERABILITY VIA INES

The Baltic case study workflow reads data from GAMS *gdx* databases and imports their data to a Spine SQL database representing all the data of the Baltic Backbone model, highlighting the capabilities of Spine Toolbox to **import and process data from varied sources**. Moreover, the model is translated to INES, to enable sharing and **interoperability with other modelling groups**, using various tools.

TOOL SPECIFICATION FRAMEWORK

DEDICATED TOOLS FOR SPECIFIC TASKS

The Baltic case study workflow, which relies on **translating the model via multiple frameworks**, achieves this through dedicated tools (INES tools), employing the ability of Spine Toolbox to incorporate in the overall workflow varied additional programs for specific tasks. This is achieved through the tool specification editor. Ultimately, the SpineOpt database can also be executed via a dedicated tool (i.e., the SpinOpt plugin).

DATA VISUALISATION AND EDITING

MODEL INSPECTION AND VALIDATION

Spine Database Editor is used extensively in the development of the Baltic case study. It allows **user-friendly and simple inspection of the model structure and parameter values**, which is essential for validation of the data importing and transformation processes. Moreover, the Graph View of Spine Database Editor allows visualizing the relationships between the various entities, also for selected subsets of the overall model.

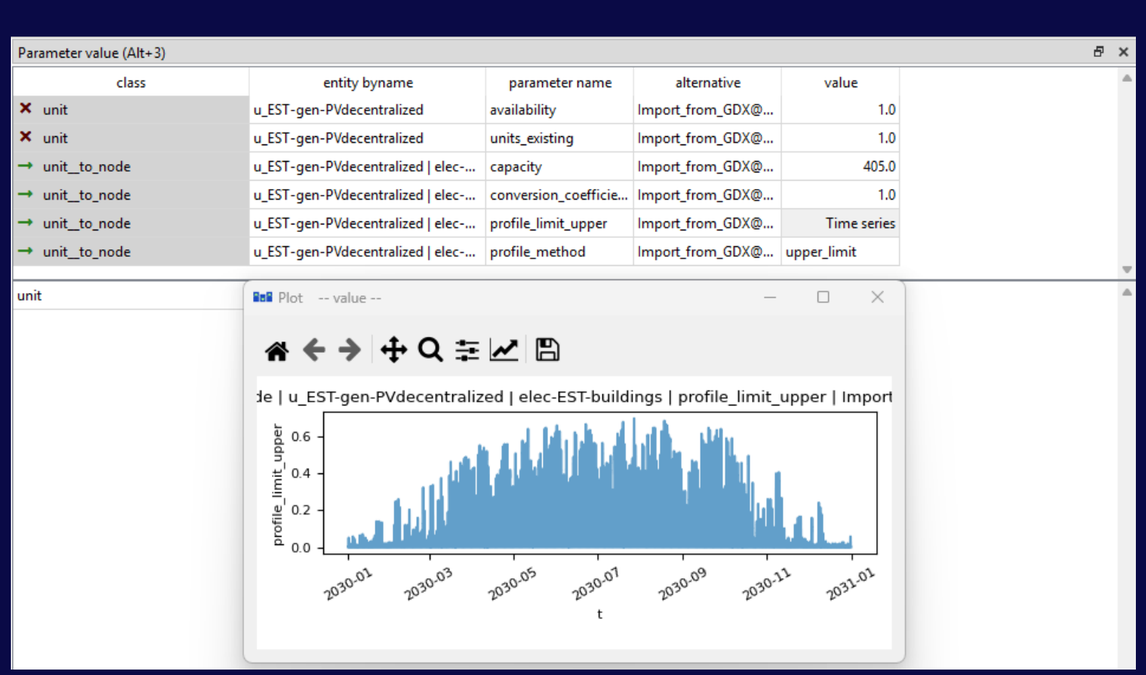


Figure 6. Graph View of two combined heat and power plants supplying heat and electricity from natural gas (INES)

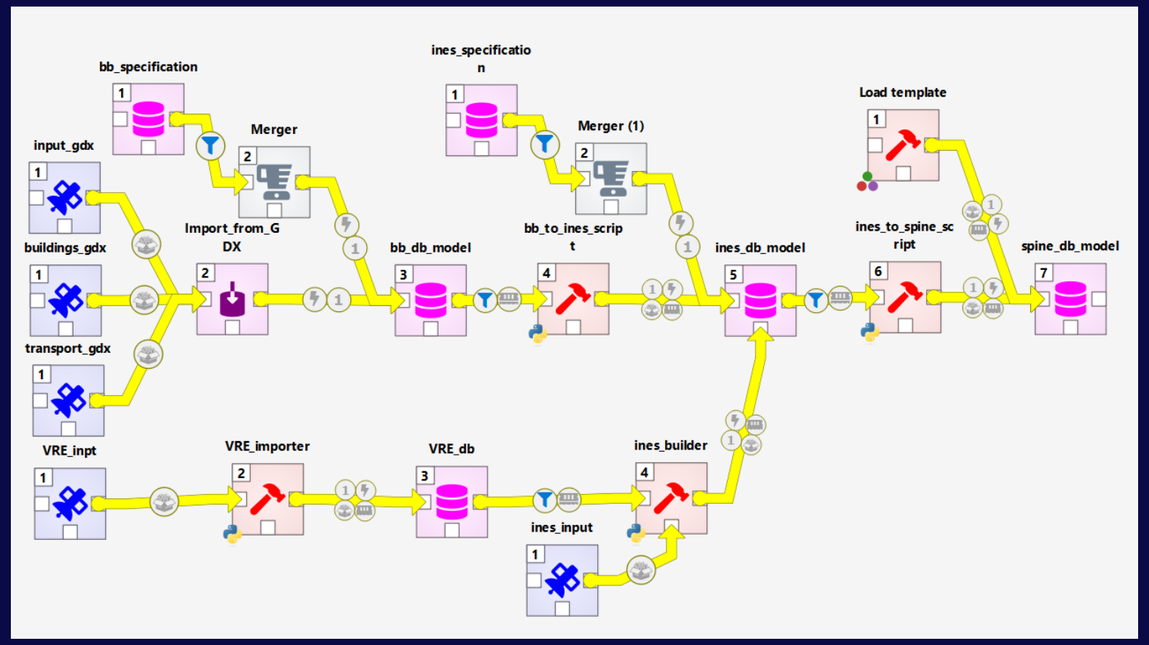


Figure 7. The Backbone -> INES -> SpineOpt transformation workflow of the Baltic case study

USE CASE 3

CWATM-FLEXTOOL

CWatM-Spine Toolbox links **detailed hydrological modelling with energy-system optimisation**. CWatM simulates water flows, reservoirs and demands, while FlexTool uses these to plan hydropower plants and energy system operations. **Spine Toolbox manages data, scenarios and workflows, enabling an integrated water-energy assessment**.

FLEXIBLE OBJECT-ORIENTED DATA MODEL

HYDROLOGICAL MODEL DATA IN SPINE DATABASE

The CWatM-FlexTool workflow relies on the Spine Data Structure to organise hydrological and energy-system inputs in a consistent way. CWatM data such as river discharge, reservoir levels and water demands are mapped into Spine entities and relationships, enabling their seamless integration into FlexTool. This **object-oriented structure allows the model to be adapted to different catchments and energy contexts without restructuring the workflow**.

ADVANCED PARAMETER HANDLING

STRUCTURED AND NATIVE MULTI-TYPE DATA

Parameters from CWatM are imported as **structured values** within the Spine database. The data consists of many datatypes in a unified environment allowing the editing and browsing of the data in a single place. FlexTool natively uses the Spine database's many datatypes in creation of the model data.

BUILT-IN SCENARIO ENGINE

DUAL WATER-ENERGY SCENARIOS

Spine Toolbox's scenario system is used to **evaluate multiple water-energy configurations**. Alternative hydrological years, climate conditions or water-use assumptions can be defined as parameter alternatives. These feed into FlexTool scenarios that explore impacts on hydropower generation, system flexibility and supply adequacy. The **scenario engine enables systematic comparison of different hydrological-energy futures**.

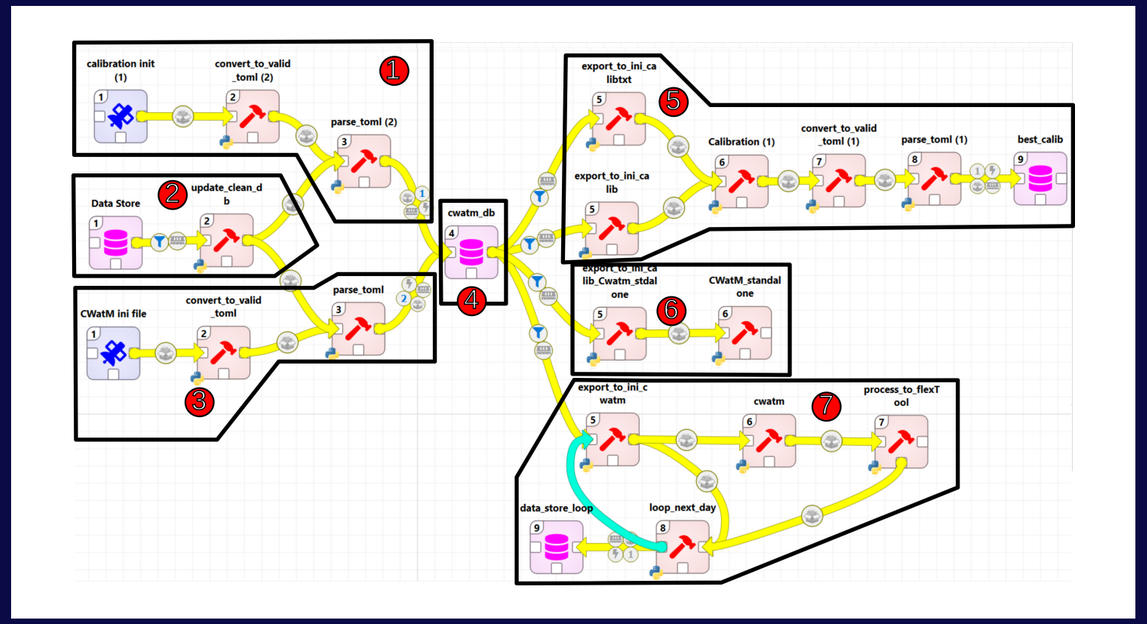


Figure 8. Spine Toolbox workflow linking CWatM hydrological simulations

DATA INTEROPERABILITY

AUTOMATED DATA CONVERSION BETWEEN THE MODELS

CWatM outputs in netCDF or text formats (TOML) are **converted into structured Spine SQL databases through dedicated importers**. FlexTool reads these transformed datasets directly from Spine, avoiding manual conversion steps. The workflow facilitates back-and-forth data integration while ensuring compatibility across tools.

TOOL SPECIFICATION FRAMEWORK

CONFIGURABLE MODEL COUPLING

The workflow incorporates CWatM execution tools and FlexTool optimisation tools using formal tool specifications. These define the input files, execution order, and generated outputs, ensuring coordinated runs of both models. The specification framework enables **automated calibration workflows, repeated model runs under alternative assumptions, and transparent orchestration of hydrological-energy model coupling**.

USE CASE 4

WEB INTERFACE APPLICATION

This use case demonstrates how Spine Toolbox can support the **development of a web-based interface for energy system optimisation workflows**. The application allows users to edit input data, define scenarios, execute selected parts of the workflow and visualise results directly in a web browser, while Spine Toolbox manages the underlying data, workflow logic and execution processes. **Spine Toolbox acts as the backend engine**, its **modular architecture** enables the separation between user interaction (web interface) and computational processes, supporting both local and remote execution.

FLEXIBLE OBJECT-ORIENTED DATA MODEL

CONSISTENT DATA REPRESENTATION ACROSS COMPONENTS

The workflow relies on Spine Data Structure to organise energy-system inputs in a consistent way **enabling clear representation of entities, relationships, and parameters** across different components of the workflow.

DATA INTEROPERABILITY

FROM RAW INPUTS TO FINAL RESULTS

Input raw data from **multiple formats** (e.g. CSV, JSON, Excel) are imported into an SQLite database using Spine Toolbox's **Importer** items, Julia Tools, and Spine Database API. The *Optimize* project item writes the results into an **output db Data Store (SQLite file)**, which is then passed on to the *Extract results* Tool. **The results are then written into an Excel file**.

TOOL SPECIFICATION FRAMEWORK

INTEGRATION OF JULIA TOOLS

Julia tools in the workflow include the SpineOpt optimization model and use-case specific data conversion tools which are integrated through **tool specifications defining their inputs, outputs, and execution logic**.

PARALLEL AND REMOTE EXECUTION ENGINE

SUPPORT FOR LOCAL AND REMOTE EXECUTION

The web app offers users the ability to execute different parts of the workflow (e.g., data preparation and optimization). The user has the option to **execute the workflow locally**, meaning that the backend container executes the project in headless mode, or the user can request that the project is **executed on a remote Spine Engine server**.

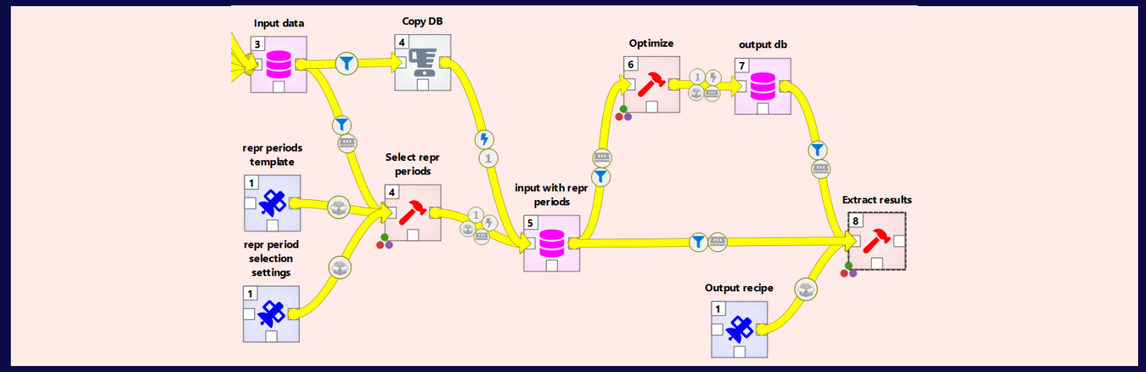


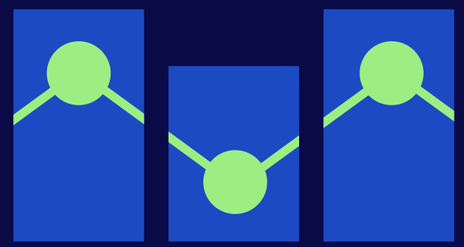
Figure 9. An example workflow executed via the web-based interface. The items downstream from input data consist of data connections containing raw data files and tools and importers for importing the data into an SQLite database in Spine format.

CONCLUSIONS

Spine Toolbox provides a **flexible and modular framework for managing data, workflows, and scenarios in complex modelling environments**. Its integration of a structured data model, workflow orchestration, and execution capabilities enables consistent handling of heterogeneous data sources and tools. Interoperability features and API access support **integration with external ecosystems**, while scenario and data management enable **systematic model analysis**. The presented use cases demonstrate its adaptability across domains and scales, supporting reproducibility, transparency, and collaborative development.

REFERENCES

Kiviluoma Juha, Pallonetto Fabiano, Marin Manuel, Savolainen Pekka T., Soininen Antti, Vennström Per, Rinne Erkka, Huang Jiangyi, Kouveliotis-Lysikatos Iasonas, Ihlemann Maren, Delarue Erik, O'Dwyer Ciara, O'Donnel Terence, Amelin Mikael, Söder Lennart, and Dillon Joseph. 2022. "Spine Toolbox: A flexible open-source workflow management system with scenario and data management" SoftwareX, Vol. 17, 100967, https://doi.org/10.1016/j.softx.2021.100967



Spine Toolbox

Open-source workflow tool for flexible scenario and data management

Authors:

Technical content: Antti Soininen, Niina Helistö, Jean-Nicolas Louis, Jussi Ikäheimo, Arttu Tupala, Pekka Savolainen (VTT), Álvaro Porras (EPRI Europe), Diāna Žalostība, Anna Mutule (RTU)

Texts: Erika Novellini (ICONS)

Graphic Design: Nicoletta Gomboli (ICONS)

TOOLS-FOR-ENERGY-SYSTEM-MODELLING.ORG

TOOLS FOR ENERGY SYSTEM MODELLING

@MOPOPROJECT

MOPO

MOPOPROJECT

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.