3rd NFDI4Energy Conference: Defining Energy Research Data Management Together

Proceedings to the 3rd NFDI4Energy Conference - Full Papers

https://doi.org/10.52825/ocp.v9i.3295

© Authors. This work is licensed under a Creative Commons Attribution 4.0 International License

Published: 23 Mar. 2026

Insights on Modelling Practices and Tool Preferences

Results of a Public Survey in the Energy Domain

Christoph Schimeczek[1,*], Felix Nitsch[2], and Daniel Zapf[1]

[1]German Aerospace Center (DLR), Institute of Networked Energy Systems, Germany

[2]BOKU University, Institute of Sustainable Economic Development, Austria

*Correspondence: Christoph Schimeczek, Christoph.Schimeczek@dlr.de

Abstract. We present results of a public survey on modelling practices and tool preferences, in the energy domain. The survey covered experiences with and usage of programming languages and models, model selection criteria, software development practices, and model interoperation. Our findings are contrasted by an assessment of openly available energy tools.

Keywords: Energy, Modelling, Tool, Preference, Practice, Survey

1. Introduction

Open modelling efforts have existed for many years in the energy domain [1] with more and more energy models going open source: [2] currently lists 210 openly available models. Based on the number of unaddressed challenges in the energy modelling domain [3] it can be expected, that even more models are to be developed. With a multitude of models to choose from in the energy landscape [4], modellers need comparative frameworks [5] to select a suitable set of tools for a given task. Applying such models and combining them becomes increasingly complex especially if models aren't designed with understandability and cooperation in mind [6]. Although there are guidelines for software development in the energy sector [7], an assessment of actual modelling practices and tool preferences is missing. This work addresses this gap by presenting results from a survey among software developers in the energy domain.

2. Survey Design

The survey was conducted from October 13th 2025 to November 10th 2025. It was publicly available and disseminated by the authors through social media, the openmod forum [8], and mailing lists of other networks, e.g., [9], and [10]: Thus, the survey does not necessarily cover a representative number of people working in the energy modelling domain. Responses to the survey were completely anonymized to lower participation barriers. 155 participants submitted at least one answer and 108 participants fully completed the survey. The structure of the survey was subdivided into five sections: 1) participant background, 2) modelling tools and programming languages, 3) model selection criteria, 4) software development practices, and 5) model interoperability. In an additional section, participants could provide free text suggestions. Most questions were mandatory. Thus, participants could not proceed in the survey without answering these questions, probably increasing the drop-out rate. Our survey design did not include techniques to identify non-meaningful responses, e.g., survey time measuring, or control questions. Thus, any findings shown in this study are rather qualitative and should be taken with a grain of salt. In the following, we present selected insights obtained from the survey.

3. Survey Results

3.1 Participant Background

The first section of the survey explored the role and level of experience of the participants with five questions, of which we discuss four herein. Question 1 asked *"What is your primary role?"*, with the options: *"Academic researcher"*, *"Industry analyst / Consultant"*, *"Software developer / Research software engineer"*, *"Government / Policy analyst"*, *"Student / PhD student"*, or *"Other"*. Of the 155 responsive participants, 78 (50%) stated to have the role of academic researcher. 27 (17%) claimed to be students, 25 (16%) identified as industry analyst or consultant. 3 stated to be associated with the role of government or policy analysts. The majority of the respondents is thus associated with academia (researchers or students). A visualisation of these shares is shown in Figure 1.

For a better understanding of the level of experience, question 2 asked *"How many years of experience do you have in energy systems analysis?"*. Participants had to choose one option from *"Less than 1 year"*, *"1-3 years"*, *"3-5 years"*, *"5-10 years"*, *"More than 10 years"*. Of the 155 responsive participants, 46 (30%) stated to have between five and ten years of experience, 37 (24%) claimed to have three to five years of experience, 33 (21%) state to have between one and three years of experience, 30 (19%) declared to have more than ten years of experience and only 9 (6%) disclosed to have less than one year of experience in the domain of energy systems analysis. Thus, a large portion of the respondents have five years or more of experience in the mentioned domain. These shares are visualised in Figure 2.

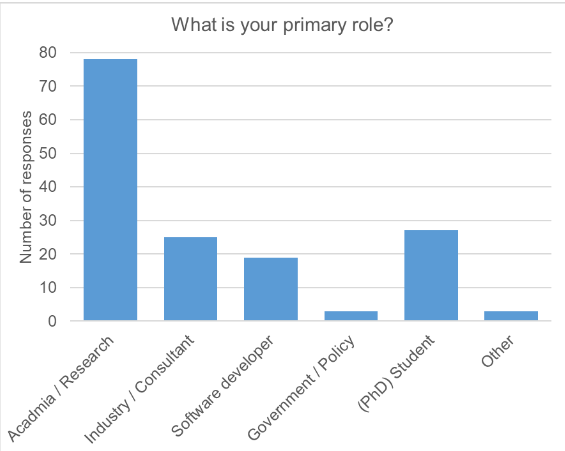

Figure 1. *Distribution of the stated primary roles of survey participants.*

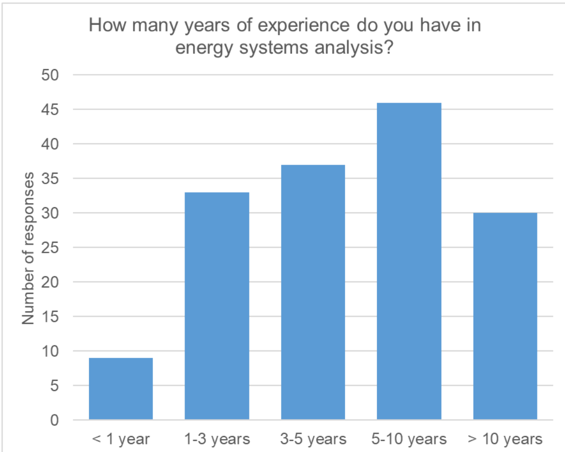

Figure 2. *Distribution of stated experience levels of the participants in the energy systems domain.*

Question 3 requested the participants to state *"How many applications, models, or repositories in the energy systems domain have you contributed code to?"*. Responding was mandatory and participants could choose from *"None"*, *"1"*, *"2-4"*, *"5-10"*, *"More than 10"*. This helps to know the technical skill assessment of the 155 participants, which responded, where 66 (43%) stated to have contributed to two to four repositories in the domain. 30 (19%) participants mentioned to have contributed to five to ten repositories, 24 (15%) claimed to have contributed to more than 10 repositories, and 13 (8%) disclosed to have not yet contribute to repositories in the requested domain. Thus, 77% of the participants stated to have contributed to at least two relevant application or model repositories. These shares are visualised in Figure 3.

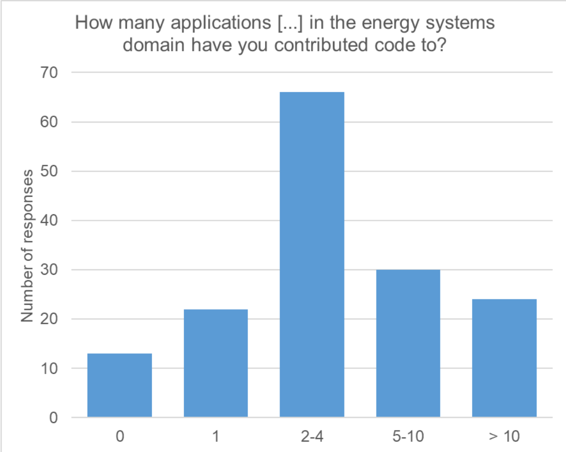

Figure 3. *Distribution of the count of repositories in the energy systems domain participants stated to have contributed to.*

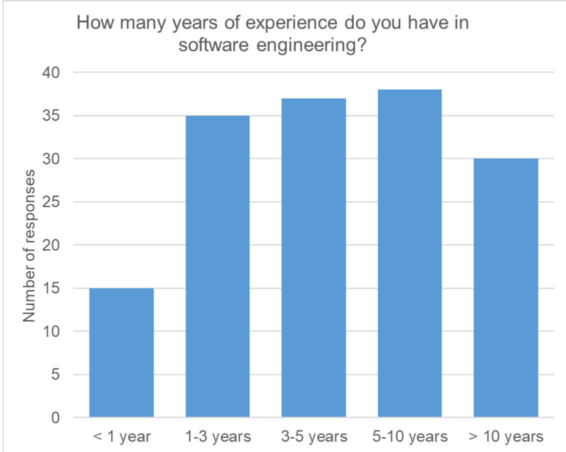

Figure 4. *Distribution of stated years of experience in software engineering.*

Question 4 asked *"How many years of experience do you have in* software *engineering?"*. Responding was mandatory. Participants could pick one response from *"Less than 1 year"*, *"1-3 years"*, *"3-5 years"*, *"5-10 years"*, *"More than 10 years"*. Again, 155 participants responded, of which 38 (25%) claimed to have five to ten years of experience, directly followed by 37 (24%) of participants stating three to five years of experience. 35 (23%) of the participants declared to have one to three years of experience, 30 (19%) mentioned more than ten years of experience and 15 (10%) disclosed to have less than one year of experience. In total, about two thirds of the participants claimed to have a least three years of experience with software engineering. The data is depicted in Figure 4.

3.2 Modelling Tools and Programming Languages

The second section of the survey explored the usage of and experience levels with programming languages and models in the energy systems domain. Five questions were asked of which three are discussed hereafter. In Question 7, *"What is your PRIMARY programming language for modelling energy systems?"*, participants could choose one of eight different programming languages from *"Python"*, *"GAMS"*, *"Julia"*, *"Rust"*, *"C/C++"*, *"Java"*, *"R"*, and *"MATLAB"*. Alternatively, participants could specify their primary language as free text. Answering this question was mandatory. 137 participants responded, of which 96 (70%) stated that Python was their primary programming language. 14 (10%) named GAMS and 10 (7%) answered Java. Julia and C/C++ was called the primary programming language by 4 (3%) participants. MATLAB and R was named by 2 (1%) each. Other mentions, each with 1 response, were Scala, RSCAD, Modelica, F#, and Excel. The responses show a clear dominance of Python as a primary programming language in the energy systems domain. A visualisation of the responses is provided in Figure 5.

In the mandatory Question 8, participants were asked "How many years of experience do you have in the following languages?". All of the abovementioned languages (*"Python"*, *"GAMS"*, *"Julia"*, *"Rust"*, *"C/C++"*, *"Java"*, *"R"*, and *"MATLAB")* were covered, while response options were *"None"*, *"Less than a year"*, *"1-2 years"*, *"3-5 years"*, *"More than 5 years"*. The question provides a trust-level of the languages. The number of responses varied slightly for the different programming languages. For Python, 137 participants responded of which 115 (84%) stated to have three years or more of programming experience (see Figure 6). Significantly less experience was found for MATLAB. 83 (61%) of the participants stated to have no experience or less than one year of experience. However, 30 (22%) declared to have at least three years of experience with this language (see Figure 7). Regarding the remaining languages, i.e., GAMS, Julia, Rust, C/C++, Java, and R, the response patters were very similar. Consistently, 70% or more of the participants claimed to have no experience or less than one year of experience. In the best case (Java), mere 16% declared to have at least three years of experience with this language. Figure 8 gives a visualisation of the results found for Java – one member of the latter group of programming languages.

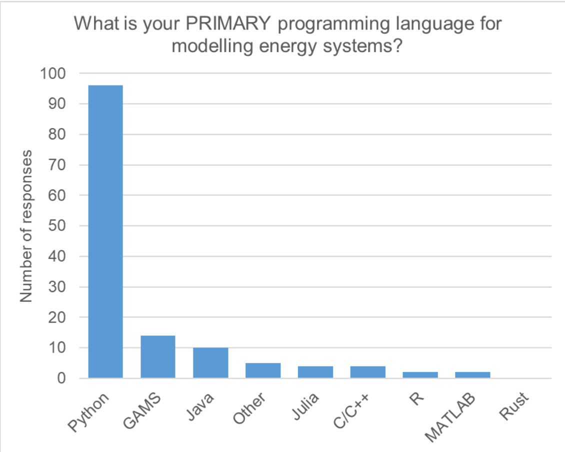

Figure 5. *Distribution of responses for the primary programming language used for modelling energy systems.*

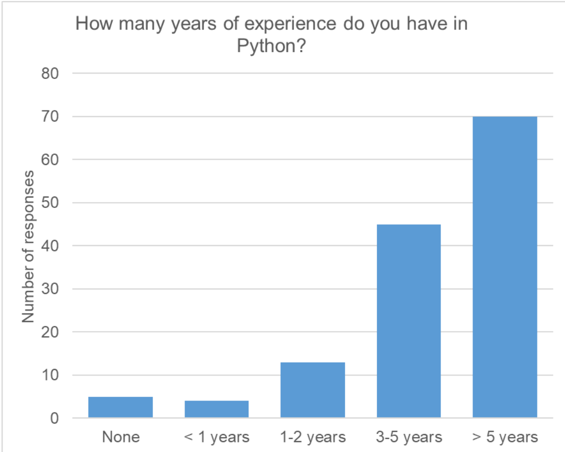

Figure 6. *Distribution of responses for the years of experience with Python.*

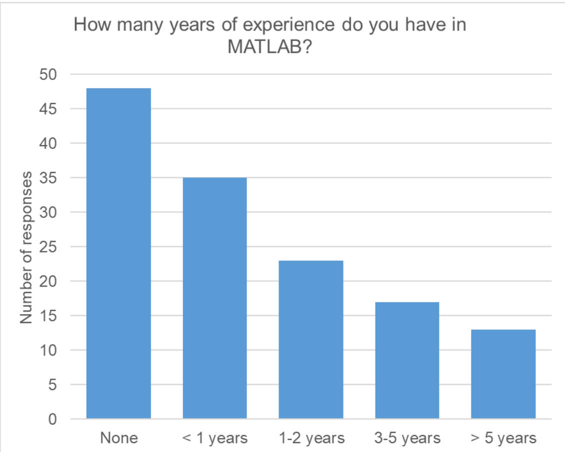

Figure 7. *Distribution of responses for the years of experience with MATLAB.*

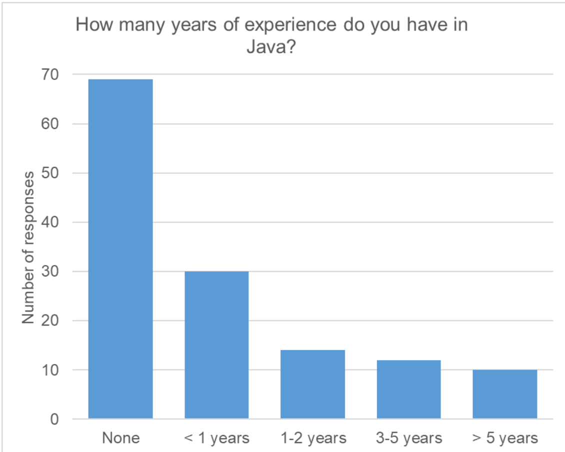

Figure 8. *Distribution of responses for the years of experience with Java.*

Question 9 asked *“Which energy systems modelling frameworks / tools do you currently use?”*. A list of seven prominent open energy models was given, namely, *“PyPSA”*, *“Pandapower”*, *“OSeMOSYS”*, *“Calliope”*, *“oemof”*, *“PLEXOS”*, and *“TIMES”*. Participants could additionally choose from *“Custom / in-house model“* or *„Other”*. Answering was mandatory, but multiple answers could be provided. It was included to understand the correlation of programming languages and the used model options. In total, 206 responses were collected from 137 participants. Results are visualised in Figure 9. Of the specified models, PyPSA was named 36 times, followed by oemof (28 mentions) and pandapower (15 mentions). 11 other models had between two and 6 mentions, and further 29 models with 1 mention were named. The largest group of 66 responses, however, were reported for custom / in-house models.

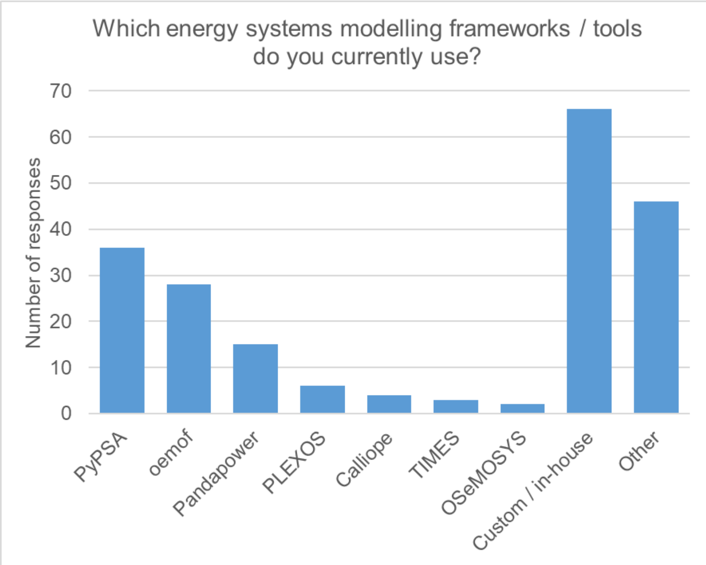

Figure 9. *Distribution of responses for used energy system models or tools currently used by the participants.*

3.3 Model Selection Criteria

The third section of the survey covered the importance of different criteria to participants in the context of model selection. This section contained a single question *"How important are the following factors to you when choosing a modelling tool?".* Participants had to choose one response from *"Not important"*, *"Somewhat important"*, *"Important"*, *"Very important"*, and *"Critical"*. Hereafter, we investigate responses for the criteria *"Open-source availability"*, *"Your knowledge of the used programming language"*, *"Computational performance"*, and *"Industry / academic acceptance"*. The total number of responses was 114 for every criterion. To ease the comparison of the figures, their y-axis scaling was aligned.

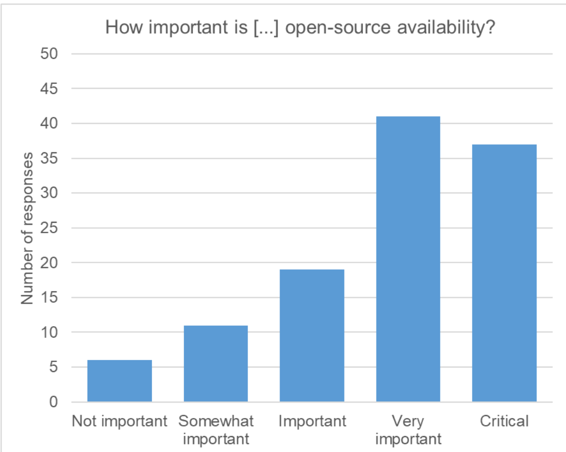

Figure 10. *Distribution of responses for the importance of open-source availability in the context of model selection.*

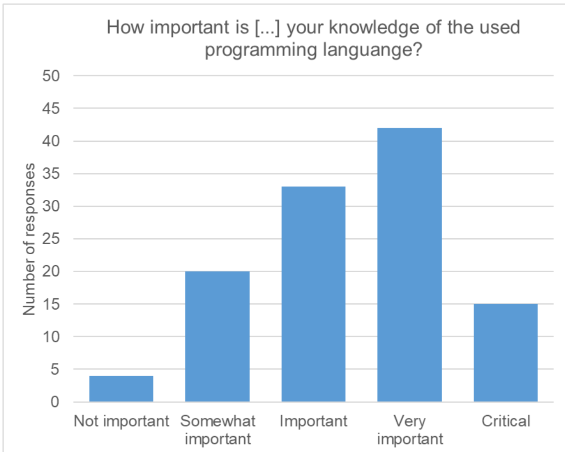

Figure 11. *Distribution of responses for the importance of programming language knowledge in the context of model selection.*

Figure 10 shows that open-source availability was rated very important or critical by 78 (68%) of the participants. Furthermore 17 (15%) participants regarded this aspect as not or only somewhat important. In case of programming language knowledge, 57 (50%) participants rated this aspect as very important or critical, whereas 24 (21%) participants deemed this aspect as not or only somewhat important (see Figure 11Figure 10). Computational performance was rated as very important or critical by 67 (59%) participants. Only 14 (12%) participants regarded this aspect as not or only somewhat important (see Figure 12). The aspect of tool acceptance in industry and academia, polarised the most. While 41 (36%) participants answered to regard this aspect very important or critical, 44 (39%) of the participants regarded this aspect as not or only somewhat important (see Figure 13).

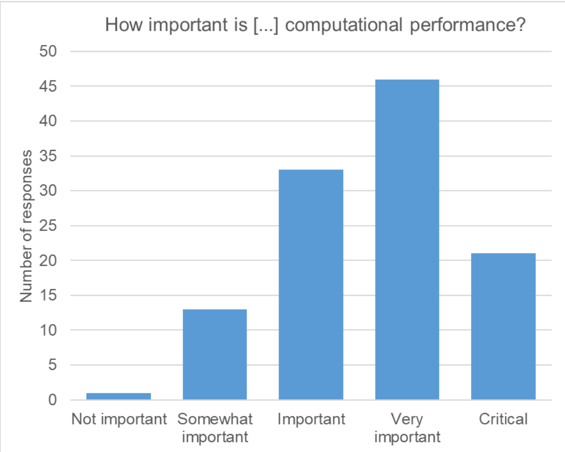

Figure 12. *Distribution of responses for the importance of computational performance in the context of model selection.*

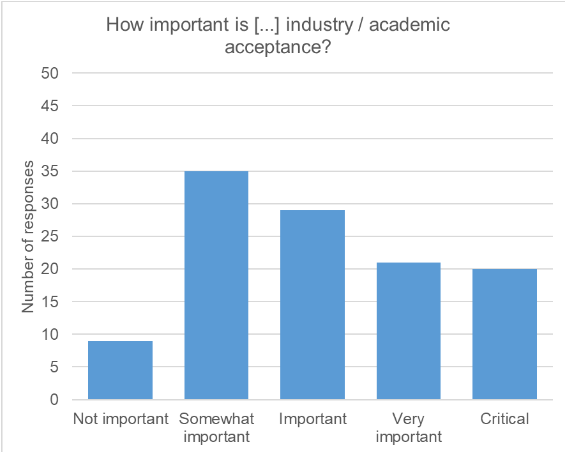

Figure 13. *Distribution of responses for the importance of industry / academic acceptance in the context of model selection.*

3.4 Software Development Practices

The last section of the survey discussed in this work covered software development practices. Three questions were asked, of which two are covered hereafter. Question 14 asked *"Do you collaborate with others for coding / scripting?"*. Participants had to choose one answer from the following options: *"Yes, regularly"*, *"Yes, occasionally"*, *"No, but it might be useful"*, *"No, I do things on my own"*. In total, 111 participants responded, of which 11 (10%) chose a type of "No", and the remainder of 100 (90%) answered with a type of "Yes". Cooperation with respect to software development is thus not uncommon in the energy domain. The results are visualised in Figure 14.

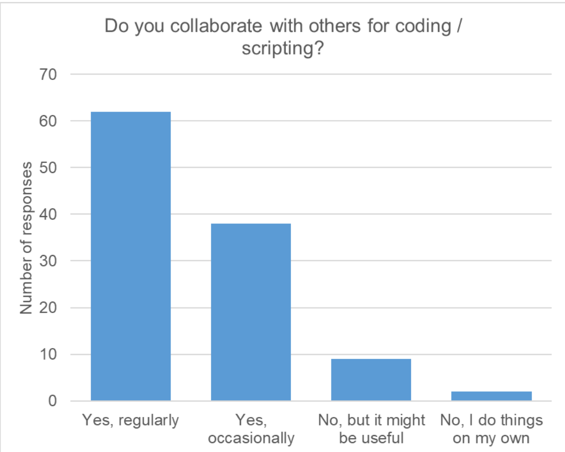

Figure 14. *Distribution of responses regarding collaboration at coding or scripting.*

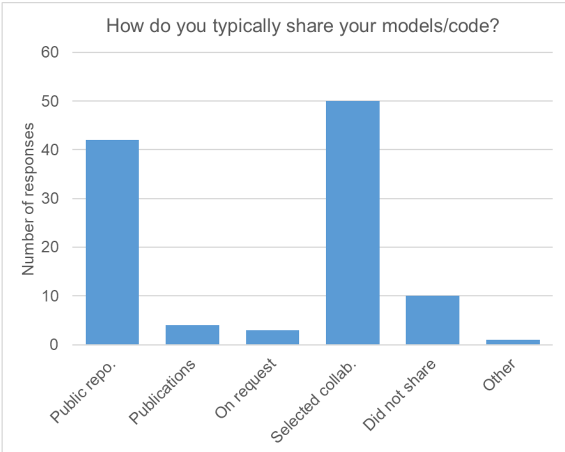

Figure 15. *Distribution of responses regarding sharing of models or code.*

In Question 16 participants were asked “How do you typically share your models/code?”. They had to choose one answer from “Public repository”, “Private repository with selected collaborators”, “Through publications (supplementary materials)”, “On request”, “I did not share, yet”, and “Other”. 110 Participants responded, of which 52 (47%) stated to typically share only with selected collaborators, but 42 (38%) answered to typically share in public repositories. A clear minority of 10 (9%) participants did not share code yet, whereas 4 (4%) and 3 (3%) of the participants typically share code via publication supplementary materials or on request.

4. Open Energy Models

To contrast the survey findings about programming language preference, we also evaluated repositories listed on the recently published openmod tracker [2]. This website collects data about open-source energy modelling tools. Data was gathered on October 16th, 2025. Of the approximately 210 listed repositories we selected those which had at least one commit in the year 2025 and extracted the primary programming language and repository stars. If any of these metrics was missing, we performed a manual lookup in the corresponding repository.

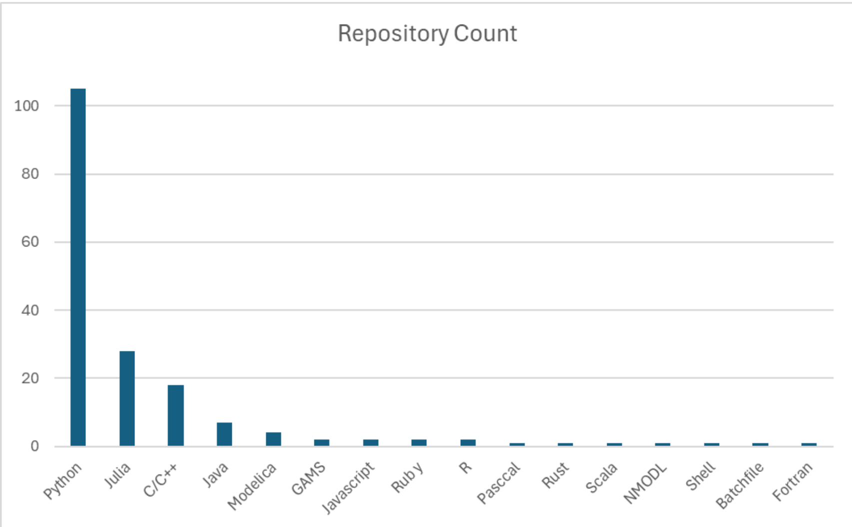

Figure 16. *Number of active repositories per primary programming language.*

Figure 16 illustrates the number of repositories associated with the programming languages in the dataset. It shows 105 (59%) of the repositories have Python as their primary programming language, followed by 28 (16%) repositories associated with Julia, 18 (10%) mainly employing C or C++, 7 (4%) repositories primarily using Java, and 4 (2%) repositories using Modelica. None of the other programming languages were found on more than 2 repositories, including GAMS, R, and Rust.

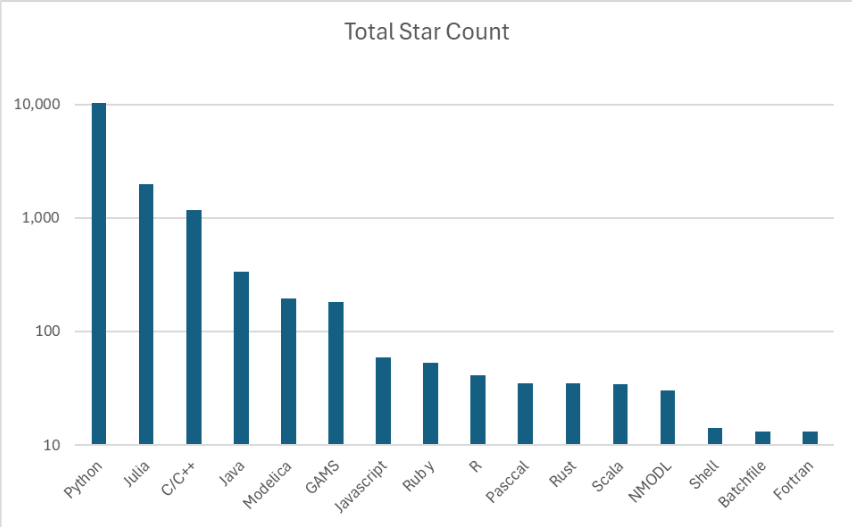

Figure 17. *Number of stars associated with repositories that have the given primary programming language.*

The situation is even more extreme, if the stars associated with the corresponding repositories are summed up and evaluated with regard to the corresponding programming language. Figure 17 depicts the corresponding results – mind the logarithmic scaling. Open-source model repositories using Python collect 71% of the stars, followed by Julia (14%), and C/C++ (10%). Java, Modelica, and GAMS repositories have a combined star share of less than 5%. All remaining programming languages muster a combined star share of approximately 2%.

5. Conclusion

Survey participants were primarily associated with academia. A significant majority of participants stated to have at least 3 years of experience in the energy domain and in the domain of software development, having contributed to at least 2 model or tool repositories in the energy domain. A share of 70% of the participants stated to primarily use Python in their work. This result matches with 71% of stars from open energy model repositories being associated with Python. This is also in line with the significantly higher levels of experience found for Python compared to the other programming languages. Julia and C/C++ were underrepresented in our survey (3% each) when compared to their repository star shares of 14% and 10%, respectively. GAMS (10%) and Java (7%) were overrepresented in our survey compared to the corresponding star share of 1% and 2%, respectively. This could be caused by the non-representative sample of the survey or a lower usage of GAMS and Java in open-source models. With regard to model selection, participants stated a high importance for open-source availability of models and model performance. A vast majority of the participants collaborate with others at coding, but only 44% share their code over public channels, whereas 56% do not share their code publicly. As this study covers only a portion of the survey data, further analyses could provide more insights on the modelling practices and tool preferences in the energy domain.

Data availability statement

Data of the presented graphs is openly available [11].

Author contributions

Christoph Schimeczek: Conceptualization, Data curation, Investigation, Visualization, Writing – original draft; Felix Nitsch: Conceptualization, Methodology, Investigation, Writing – review & editing; Daniel Zapf: Conceptualization, Investigation, Writing – review & editing

Competing interests

The authors declare that they have no competing interests.

Acknowledgement

The authors express their gratitude to the survey participants for their time, as well as to Stephan Ferenz and Lea Kuhlmann for disseminating the survey. Furthermore, the authors thank Leonard Willeke for reviewing the survey design.

References

[1] S. Pfennigner et al., "Opening the black box of energy modelling: Strategies and lessons learned", Energy Strategy Reviews 19, 2018, https://doi.org/10.1016/j.esr.2017.12.002
[2] Open Energy Modelling Tool Tracker. https://openmod-tracker.org/ (accessed November 24th, 2025)
[3] F. Scheller et al., "An expert survey to assess the current status and future challenges of energy system analysis", Smart Energy 4, 2021, https://doi.org/10.1016/j.segy.2021.100057
[4] LF Energy Landscape. https://lfenergy.landscape2.io/ (accessed November 24th, 2025)
[5] F. Wiese et al., "A qualitative evaluation approach for energy system modelling frameworks", Energy, Sustainability and Society 8, 2018, https://doi.org/10.1186/s13705-018-0154-3
[6] S. Pfenninger, "Open code and data are not enough: understandability as design goal for energy system models", Progress in Energy 6(3), 2024, https://doi.org/10.1088/2516-1083/ad371e
[7] S. Ferenz et al., "Ten Recommendations for Engineering Research Software in Energy Research", E-Energy '25 Proceedings, 2025, https://dl.acm.org/doi/10.1145/3679240.3734606
[8] openmod Forum. https://forum.openmod.org (accessed November 20th, 2025)
[9] Strommarkttreffen. https://www.strommarkttreffen.org (accessed November 20th, 2025)
[10] NFDI4Energy. https://nfdi4energy.uol.de (accessed November 20th, 2025)
[11] C. Schimeczek, F. Nitsch, D. Zapf, "Insights on Modelling Practices and Tool Preferences (Dataset)", Zenodo, 2025, https://doi.org/10.5281/zenodo.17659978