Autonomous battery optimisation by deploying distributed experiments and simulations

Monika Vogler[1,2,11], Simon K. Steensen[4,11], Francisco Fernando Ramirez[5], Leon Merker[1], Jonas Busk[4], Johan M. Carlsson[6], Laura Hannemose Rieger[4], Bojing Zhang[1,3], François Liot[5], Giovanni Pizzi[7,5], Felix Hanke[8], Eibar Flores[9], Hamidreza Hajiyani[6], Stefan Fuchs[1], Alexey Sanin[1,3], Miran Gaberšček[10], Ivano E. Castelli[4], Simon Clark[9], Tejs Vegge[4], Arghya Bhowmik[*, 4], and Helge S. Stein[*, 2]

[1]Helmholtz-Institut Ulm (HIU), Helmholtzstr. 11, 89081 Ulm
[2]Technical University of Munich, Germany; TUM School of Natural Sciences, Department of Chemistry, Chair of Digital Catalysis; Munich Institute of Robotics and Machine Intelligence (MIRMI); Munich Data Science Institute MDSI
[3]Present address: Technical University of Munich, Germany; TUM School of Natural Sciences, Department of Chemistry, Chair of Digital Catalysis; Munich Institute of Robotics and Machine Intelligence (MIRMI); Munich Data Science Institute MDSI
[4]Technical University of Denmark (DTU), Department for Energy Conversion and Storage, 2800 Kgs. Lyngby, Denmark
[5]Theory and Simulation of Materials (THEOS) and National Centre for Computational Design and Discovery of Novel Materials (MARVEL), École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland
[6]Dassault Systèmes Germany GmbH, Am Kabellager 11-13, D-51063 Cologne, Germany
[7]Laboratory for Materials Simulations (LMS), Paul Scherrer Institut (PSI), CH-5232 Villigen PSI, Switzerland
[8]Dassault Systèmes, 22 Science Park, Cambridge CB4 0FJ, UK
[9]SINTEF Industry, Battery Technology, 7034 Trondheim, Norway
[10]Department of Materials Chemistry, National Institute of Chemistry, Hajdrihova 19, 1000, Ljubljana, Slovenia
[11]M. Vogler and S. K. Steensen contributed equally.
[*]Corresponding authors.

May 14, 2024

Non-trivial relationships link individual materials properties to device-level performance. Device optimisation therefore calls for new automation approaches beyond the laboratory bench with tight integration of different research methods. We demonstrate a Materials Acceleration Platform (MAP) in the field of battery research based on our problem-agnostic Fast INtention-Agnostic LEarning Server (FINALES) framework, which integrates simulations and physical experiments without centrally controlling them. The connected capabilities entail the formulation and characterisation of electrolytes,

cell assembly and testing, early lifetime prediction, and ontology-mapped data storage provided by institutions distributed across Europe. The infrastructure is used to optimise the ionic conductivity of electrolytes and the End Of Life (EOL) of lithium-ion coin cells by varying the electrolyte formulation. We rediscover trends in ionic conductivity and investigate the effect of the electrolyte formulation on the EOL. We further demonstrate the capability of our MAP to bridge diverse research modalities, scales, and institutions enabling system-level investigations under asynchronous conditions while handling concurrent workflows on the material- and system-level, demonstrating true intention-agnosticism.

1 Introduction

The slow process of discovering new materials with enhanced properties, long term durability and ease of device integration is the critical bottleneck in the green transition. Automated materials discovery and optimisation mark a breakthrough in this direction often captured by the general concept of a self-driving laboratory (SDL) [1] and in a broader context as Materials Acceleration Platforms (MAPs) [2, 3]. This approach involves autonomous or closed-loop experimentation, integrating robotics and Artificial Intelligence (AI) to autonomously conduct, assess, and steer experiments using data analysed through machine learning (ML). Such platforms transcend conventional lab automation by solving problems without the need for human interaction.

Conventional studies in battery research focus on the optimisation of a preselected set of materials properties before finally testing the optimised materials in cells. Due to the multitude of materials and interfaces in battery cells, this Edisonian one-variable-at-a-time method makes the discovery of new materials for high-performing batteries a time and resource intensive task, because the characteristics of the systems are not merely the aggregate of the components. For example, the performance and degradation resistance of a device (e.g. a battery) depend on the interplay of its constituent materials and resulting interfaces [4, 5]. Materials discovery therefore requires an optimisation across all time and length scales, as well as among all involved materials. Directly targeting the optimisation of the system has the potential to significantly shorten the time-to-market, while still gaining knowledge about the complex interplay between the materials and the interfaces.

Combining accelerated autonomous discovery capability with a device level perspective has very recently been introduced for opto-electronic systems [6], but has been out of reach for electrochemical system discovery. Electrochemical materials discovery and development is a complex, multi-pronged process spanning prediction, synthesis, characterisation, and device testing processes. Related critical research tasks often hinge on the access to bespoke equipment, e.g., AI-supercomputers for generative design of new materials using foundation models [7], or high-fidelity synthesis and characterisation equipment in large-scale facilities [8]. Such resources often have limited and delayed access that requires peer-reviewed applications for compute resources or beamtime. Com-

bining such infrastructure within a MAP requires a novel direction in MAP architecture in terms of distributed operation and decentralisation by design. Surpassing pure acceleration [9] and establishing reliability of discovery [10], such a MAP will encourage the thorough characterisation and testing of full systems rather than individual materials or components. This also provides dual benefits of higher throughput and better reproducibility by removing human error and providing improved automated documentation of experiments including negative results. The practice of publishing complete datasets from SDL operation alongside the adoption of standardised machine readable data formats, holds the promise of boosting data reusability and, consequently, further resource efficiency. By leveraging carefully designed autonomous experiments in conjunction with experiment planning algorithms, it is possible not only to reduce material usage but also enhance efficiency by minimising downtime of the instruments.

To address the aforementioned challenges, we discuss here the versatile and ontology-linked Fast INtention-Agnostic LEarning Server (FINALES) framework capable of providing Application Programming Interface (API)-access to a distributed MAP enabling fully autonomous operation. The use of clearly defined data structures permits a unified and unambiguous communication without the need for human intervention. Since the framework passively enables the communication without actively triggering actions within the MAP, new clients (tenants) may be added without necessitating alterations in the internal processes and workflows. This agile construct leads us to expect facile scalability of MAPs based on the FINALES framework. Integrating patterns [11] and automated analysis frameworks like the Modular and Autonomous Data Analysis Platform (MADAP) [12] empowers even non-expert users to deploy the capabilities offered in the MAP. The link to the BattINFO [13, 14] ontology enables the datasets to be found by semantic searches while the connection to the BIG-MAP Archive [15] ensures a timely dissemination of the generated data using similar technology as the Materials Cloud Archive [16].

In this study, we demonstrate an internationally distributed MAP orchestrated by FINALES as shown in Figure 1 working on two independent optimisation tasks in parallel. Both optimisation tasks vary the composition of a battery electrolyte composed of ethylene carbonate (EC), ethyl methyl carbonate (EMC), and lithium hexafluorophosphate ($LiPF_6$), but one targets the optimisation of the ionic conductivity, while the other aims to maximise the End Of Life (EOL) of coin cells. We showcase the hierarchical acceleration in complex device level MAPs, both by intelligent sampling of the design space (here the chemical space of the electrolyte) as well as accelerating individual slow and resource-intensive tasks (here EOL testing) with ML. We further demonstrate the use of complementary methods for determining the ionic conductivity enlarging the chemical space accessible to the MAP as a whole. Moreover, we show the multi-tasking capability of our MAP by running the two optimisation tasks in parallel on the same infrastructure. The setup was able to discover the correlation between the concentration of the salt and the ionic conductivity of the electrolyte and showed reasonable agreement between predicted and experimentally determined median EOL.

To perform the optimisations, the following tenants were registered in FINALES with capabilities stated in parenthesis: ASAB tenant (formulates electrolytes and measures

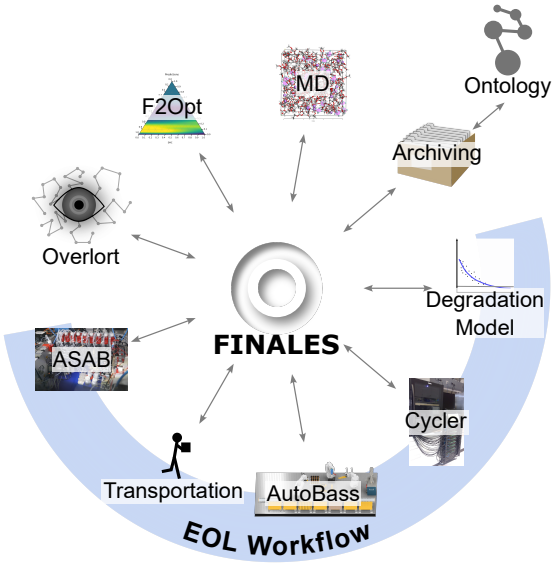


Figure 1: The layout of the MAP centred around FINALES. The tenants involved in the workflow related to the optimisation of the EOL orchestrated by the OVERLooking ORchestrating Tenant (Overlort) are spanned by the blue arc. The communication is realised exclusively via FINALES.

ionic conductivity), Molecular Dynamics tenant (estimates ionic conductivity), AutoBASS tenant (assembling coin cells), Cycler tenant (battery cycling), Degradation model tenant (early lifetime prediction), Transportation tenant (transportation of physical samples), Overlort (workflow management), F2Opt (active learning optimiser). Additionally, the Archiving tenant (archiving results) interacted with FINALES. Further details on the tenants are provided in section 4.3.

2 Results and Discussion

The study consist of two phases. In the *single-task* phase, only the tenants involved in the optimisation of the ionic conductivity of the electrolytes are active. The *multi-task* phase starts with a new, empty database and includes the optimisation of ionic conductivity in parallel to the optimisation of the EOL with all the tenants listed in section 4.3. Further details on the phases are reported in section 4.4.

2.1 Ionic conductivity and chemical space

Figure 2 presents the chemical space accessible to the Molecular Dynamics (MD) and Autonomous Synthesis and Analysis of Battery electrolytes (ASAB) tenant including the requested formulations and the reported results. Additionally, predictions for the formulations providing maximum ionic conductivity are indicated for each of the tenants. These predictions are obtained from an optimiser model after training it on the final data

available in the database after completion of the data acquisition. The optima include a predicted global optimum (best predicted global), a predicted optimum respecting the limitations of the respective tenant (best predicted limited) and the highest mean ionic conductivity observed in the reported results (best observed). The numerical values of the identified optima are reported in Table SI-3. The top part of Figure 2 further shows a change of the limitations for the ASAB tenant, which was necessary because the initially included lower limit for EMC of 0.35 mol.− % (set based on the solubility limit reported by Ding, Xu, and Jow [17]) was practically not accessible using the stock solutions chosen for this study.

The formulations with maximum ionic conductivity are found in the same band of $LiPF_6$ concentrations for the single-task phase and the multi-task phase, suggesting that the results and insights generated by the MAP are reproducible. The reproducibility is also evident from the value of maximum ionic conductivity, which is obtained for very similar formulations repeatedly tested in the high ionic conductivity region. Also, Figure 2 shows that the chemical space that can be explored by the MAP is significantly enlarged by using the MD tenant in addition to the ASAB tenant.

It also gets clear from the top part of Figure 2 that the optimiser explored more of the chemical space for the MD tenant than for the ASAB tenant. Possibly, this is due to the significantly larger region of the chemical space being accessible to the MD tenant than to the ASAB tenant. However, it also gets obvious that the optimiser requests a lot of formulations close to the edges of the chemical space of the MD tenant. This behaviour likely occurs due to the model trained by the optimiser having a high uncertainty near the edges of the accessible chemical space. Regions with high predicted values and high uncertainty bear the potential for formulations with high ionic conductivity and are therefore sampled by the optimiser acquisition function. A larger number of randomly selected starting formulations (1 in the single-task phase vs. 10 in the multi-task phase) seems to have resulted in a better initial model and consequently improved exploration of the chemical space in the multi-task phase of the campaign as can be seen in the bottom part of Figure 2.

2.2 Ionic conductivity and EOL

A plot of the ionic conductivity *versus* the molality of $LiPF_6$ in the electrolyte formulation as shown in Figure 3a reveals an increasing ionic conductivity for increasing $LiPF_6$ concentration at low molalities. At approximately 1 m $LiPF_6$, the ionic conductivity reaches a maximum at approximately $1.0\,S\,m^{-1}$ prior to a decrease at higher molality. This observation matches with the behaviour frequently reported in the literature [18–24]. The data obtained from the empirical model published by Ding *et al.* [18] used in this study for calibration of the conductivity measurement and simulation taking the molality of the salt, the mole fraction of EC in the solvent and the temperature into account is also shown in Figure 3a. Besides, data published by Choobar *et al.* [23] and Logan *et al.* [25] is presented in the graph, which is in good agreement with the data obtained from our MAP. It needs to be mentioned that the graphs shown in Figure 3a comprise averaged ionic conductivity values for all the formulations, which are reported

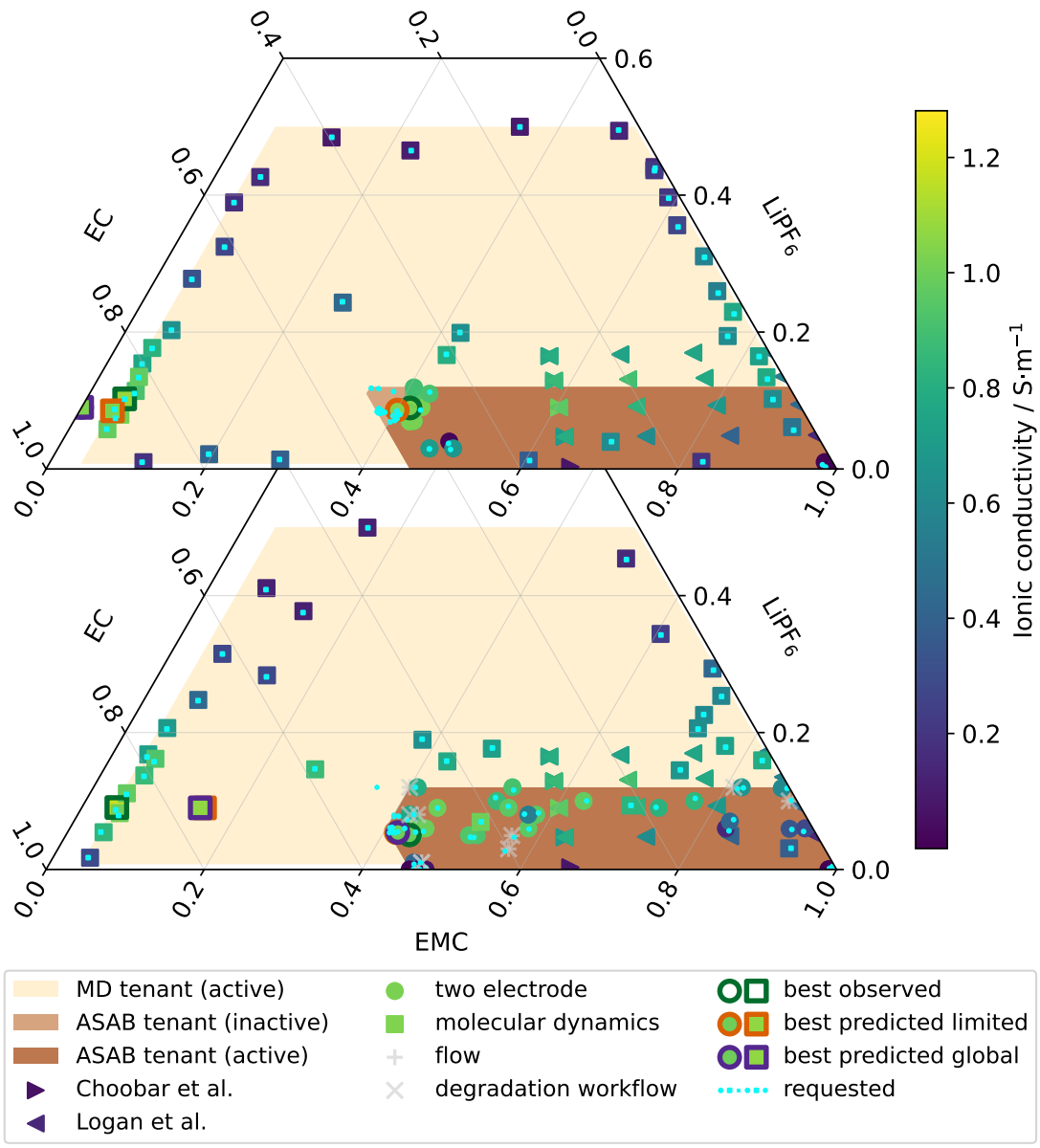


Figure 2: The chemical space accessible to the MD tenant and the ASAB tenant in the single-task phase (top) and the multi-task phase (bottom) of the study, including the formulations for which conductivity data was generated. The formulations requested in the context of the workflow associated with the EOL optimisation are included, although no conductivity data are available for those samples. The colouring of the data points is based on the mean ionic conductivity. The colourbar and legend are valid for both graphs.

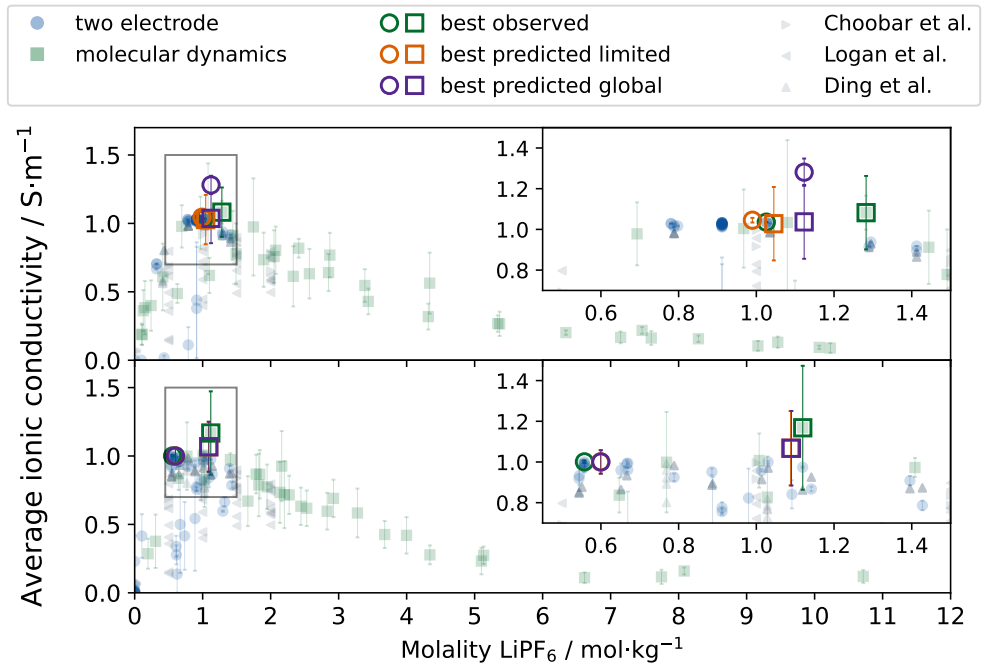
in Figure 2. Therefore, the composition of the solvent is varied besides the molality of the salt. However, the effect of changes of the salt concentration is reported to be significantly stronger than the influence of the solvent composition [18–20], which is also found in our data as visible in the Figures SI-15 - SI-18 in the Supplementary Information (SI). Variations in temperature are also not considered in Figure 3a, which is expected to cause some additional variation in the reported values for ionic conductivity. The observation of the same trend with molality in the single-task phase and the multi-task phase proves the reproducibility of the insights generated by our MAP.

Additional to the ionic conductivity, the EOL is investigated in the multi-task phase of the campaign. A second instance of the optimiser performs this fully independent optimisation task on the same hardware and software infrastructure on which the single-task phase optimisation was run previously. Since the early lifetime predictions require a complex, multi-step workflow involving hardware and software operations, the Overlort tenant is demonstrated to handle this workflow. The optimisation of the EOL was started on 13$^{\text{th}}$ of November, 2023.

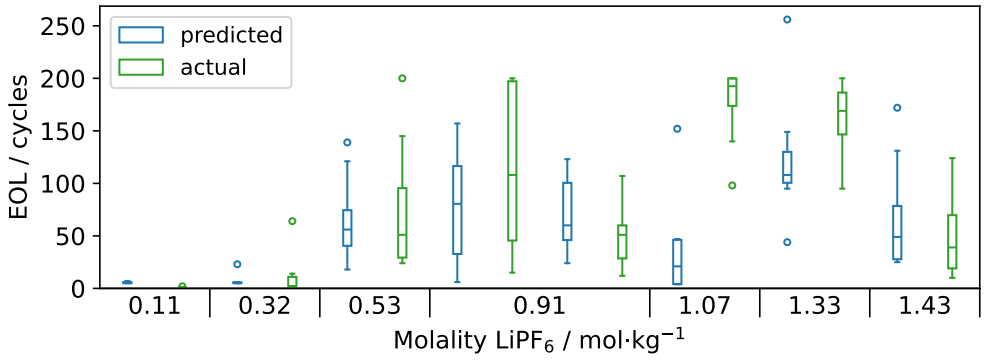
In the typical mode of operation, requests are posted by the optimiser. To investigate whether an electrolyte formulation optimised for ionic conductivity also yields an optimised EOL, additional requests are manually posted to FINALES during the multi-task phase. These manually posted requests include the formulation optimised for ionic conductivity and two non-optimised formulations as identified in the single-task phase. It must be noted that the predicted optimised formulations used for the manual requests in the multi-task phase differ from the optimised formulations reported in Figure 2 and Table SI-3, since the model is retrained after each iteration and therefore the predictions based on a model trained later differ from earlier models. The electrolyte formulations included in the manually posted requests are reported in Table SI-4.

As shown in Figure 3b, the median EOL appears to show a slight trend towards higher values for increasing molality of the electrolyte formulation. Considering the distribution of the EOL data within and in between the batches, this trend does not persist due to the strong overlap of the distributions. The maximum experimentally determined median EOL is observed for 1.07 m $LiPF_6$.

From the comparison of Figure 3a and Figure 3b, higher ionic conductivity as well as median EOL are observed for increasing molality of $LiPF_6$ until 1 m $LiPF_6$ is reached. Further increasing molality is found to decrease both quantities. The data generated in this study suggest an agreement between the two optima around 1 m $LiPF_6$. Further investigation is, however, needed to validate this observation.



(a)



(b)

Figure 3: (a) The average ionic conductivity (κ_{avg}) plotted against the molality of $LiPF_6$ as recorded during the single-task phase (top) and multi-task phase (bottom) of the campaign. Including the predicted optima globally (best predicted global) and respecting the limitations of the tenants (best predicted limited) as well as the highest observed value (best observed) for the MD and the ASAB tenant, respectively. Variations in temperature are not represented in this graph. The insets show a zoomed-in view of the region of maximum κ_{avg}. Literature data are obtained from the empirical model reported by Ding *et al.* [18] and from the publications of Choobar *et al.* [23], and Logan *et al.* [25]. (b) EOL data generated in the multi-task phase of the campaign *versus* the molality of $LiPF_6$ in the electrolytes. Note that the composition of the solvent in the electrolyte also differs between the datapoints.

3 Conclusion

In this study, we demonstrated that an internationally distributed MAP based on the FINALES framework and composed of several software and hardware tenants can run two distinct optimisation tasks while seamlessly integrating computational and experimental capabilities and bridging the gap between the material and system scales. The system was shown to reproducibly identify regions of high ionic conductivity within an electrolyte system composed of $LiPF_6$, EC, and EMC. Computational and experimental tenants connected to FINALES complemented each other, widening the chemical space accessible to the MAP as a whole. Furthermore, the communication of limitations for individual tenants, and the extensive use of identifiers and timestamps, allowed for improved data management and traceability in comparison to the previous version of FINALES [3].

Our study demonstrated ML acceleration of the electrolyte optimisation and the cycling task by early prediction of the EOL. This shows that in MAPs centred around FINALES, the overall optimisation task as well as the tasks of individual tenants can be accelerated using ML. Besides, this study demonstrated the ability of our MAP to handle tenants with significantly different rates of data generation. To accommodate these differences, it is inevitable to allow each tenant to operate at its own schedule in an asynchronous fashion. Marking requests as `reserved` once they are processed enables a distribution of the tasks among faster and slower tenants providing the same quantity. The modular design of the FINALES-based MAP leads us to expect seamless connection of new tenants and, hence, scalability to larger MAPs. We anticipate that the multi-modal and multi-fidelity nature of the data generated in such MAPs benefits the robustness of models trained with the data. Future studies deploying our MAP concept will additionally benefit from the possibility to prepopulate a database with selected existing data. This is powered by FINALES accepting unsolicited results which do not reference to a request. Starting from existing data will reduce the need for initial random experiments and therefore accelerate optimisation tasks.

We are convinced that this robust demonstration of an operating, distributed MAP centred around a passive brokering server with mostly *intention-agnostic* tenants proves our approach to be valid even for complex optimisations. Future optimisation tasks for MAPs powered by FINALES will therefore encompass multi-objective optimisations requiring the identification of Pareto-optimal solutions. These approaches shall continue following the scale-bridging approach presented in this study. Since this study also included significant efforts in designing and developing tenants, we expect a further acceleration of data generation once more mature tenants are available.

4 Methods

4.1 FINALES Brokering Framework

The FINALES architecture encompasses a server framework designed to provide communication protocols and access to a database to affiliated clients, referred to as *tenants*.

Moreover, it hosts a centralised queue for task requests enabling the tenants to pull new tasks for processing and subsequently post the results. Interactions are possible via a web server interface through the implementation of the HTTP REST API protocol. The backend offers a well-structured relational database, where all incoming requests and results are stored. Alongside the scientific data, details about connected tenants as well as information about the quantities and methods available from the MAP are saved. Further, the database holds information about the data provenance and a history of the status of tenants, methods, quantities, requests and results. *Quantities* in the sense of the FINALES framework may be measurable quantities, but can also refer to a service like e.g. the transportation of a sample. The term *method* refers to a means of providing a value for a quantity, which may be an experimental or computational procedure. Since only combinations of quantities and methods are required to be unique in the FINALES setup, this combination is referred to as a *capability*. Tenants are therefore clients that provide one or more capabilities in the MAP. This means, they regularly call the server for requests related to their associated quantity and, after applying a computational or experimental method, return a result for that quantity. FINALES does not restrict its tenants regarding their source of data or the type of their method. Experimental setups and laboratories run by humans as well as data-driven models or physics-based simulations are accepted as tenants.

The modular approach of the framework allows for facile registration of new tenants while explicitly allowing for several tenants offering the same capabilities. We refer to this concept as *multitenancy*. FINALES enables modularity and diversity of the tenants by not actively triggering actions in the MAP, but passively providing access to the data to all the tenants. Since tenants pull requests for capabilities that they can serve from FINALES, the MAP operates asynchronously and each tenant may process the requests at its own schedule. This permits multi-task operation of the MAP with several optimisers sending requests related to their respective and potentially independent tasks. Depending on the degree of automation of the tenant, a MAP based on FINALES can operate fully autonomously exceeding the limitations of a single laboratory or institution.

The communication within the MAP is based on JSON (JavaScript Object Notation) schemas, which are registered with FINALES and allow for the validation of submitted data structures. The overall schema design is developed to be problem-agnostic, allowing it to accommodate several capabilities, going beyond the specific implementation relevant for the optimisation of electrolyte formulations that we demonstrate in this study. The JSON schemas developed for this study are detailed in section 4.2.

In a typical iteration of an optimisation task, an optimiser tenant posts a request for a certain quantity. Requests must specify only one quantity but may include a list of acceptable methods. According to the multitenancy approach, several tenants may therefore be allowed to serve the request. This enables an improved use of instruments and resources as the earliest available tenant will process a request accepting its method. However, this requires the tenants to mark the request as `reserved` prior to start processing it, to hinder other tenants from picking up the same request. Once a tenant picked up the request and marked it as `reserved`, the relevant method of the tenant is executed and the results are posted to FINALES after formatting them according to the

applicable output schema. An optimiser tenant may subsequently check for new results and generate a follow-up request starting the next iteration.

An earlier proof of concept (PoC) version of the framework [3] demonstrated the basic concept of this MAP design. The present study utilises a new implementation of the concept, and applies it to an optimisation of a battery electrolyte system including the simultaneous operation of two optimiser instances. The completely redesigned framework has significantly updated communication schemas. The data structures implemented in the latest version of FINALES presented here are well defined but sufficiently generic to accommodate a variety of results for different quantities provided by diverse methods. These generic data structures are enhanced by external schemas, which are more specific to the use case of the FINALES instance. The specific schemas can be applied to the MAP by a human administrator, who registers them with FINALES and thus makes them binding for data related to the affected quantity and method. This composite design of the schemas significantly enhances the flexibility of the FINALES framework compared to the previous version, which based the communication on a single set of data structures. Outcomes from the earlier PoC implementation of FINALES [3] helped us identify the improvements needed during our reimplementation of the code. We incorporated, among other improvements, a more robust database design, enhanced scalability, and implementation of generic rigid schemas for registration and communication. The changes make the new implementation of FINALES configurable with respect to the used schemas, which even allows for hot-swaps. Compared to the previous version this is a significant improvement as it externalises the usecase-specific aspects from the framework, rendering it more universal. Additionally, the ubiquitous use of timestamps and Universially Unique Identifiers (UUIDs) in the new implementation improves the traceability of data and events in the MAP.

The repository with the source code of the redesigned FINALES framework can be found at `https://github.com/BIG-MAP/FINALES2` and the version 1.1.0 of the FINALES code used in this study is available at `https://doi.org/10.5281/zenodo.10987727`.

4.1.1 Database design

The Python-based backend of FINALES deploys a relational database for data storage, whose design is visualised in Figure 4. The code is currently set up with the SQLite [26] database engine, with all database interactions performed via SQLAlchemy [27]. Utilising SQLAlchemy allows for future deployments of the server using any of the alternative supported engines such as PostgreSQL (https://www.postgresql.org/), MySQL (https://www.mysql.com/), etc.

The database tables `quantity`, `tenant`, `request`, `result`, `link_quantity_request`, `link_quantity_result`, `status_log_request` and `status_log_result` depicted in Figure 4 all have a `uuid` column as a primary key, and `load_time` column time-stamping when the row is first appended to the table. This design, along with a backend logger, ensures traceability and reproducibility, which facilitates an exact tracing of the sequence of operations performed in the MAP at any point in time. The tables adhere to an

append-only design, with only a few exceptions controlled through `UPDATE` interactions, e.g., for fields defining whether specific quantity and tenant entries are currently active in the MAP. A small expansion of the database schema allowing for a strict append-only design can be found in Figure SI-1, which will be implemented in the release of version 2.0.0.

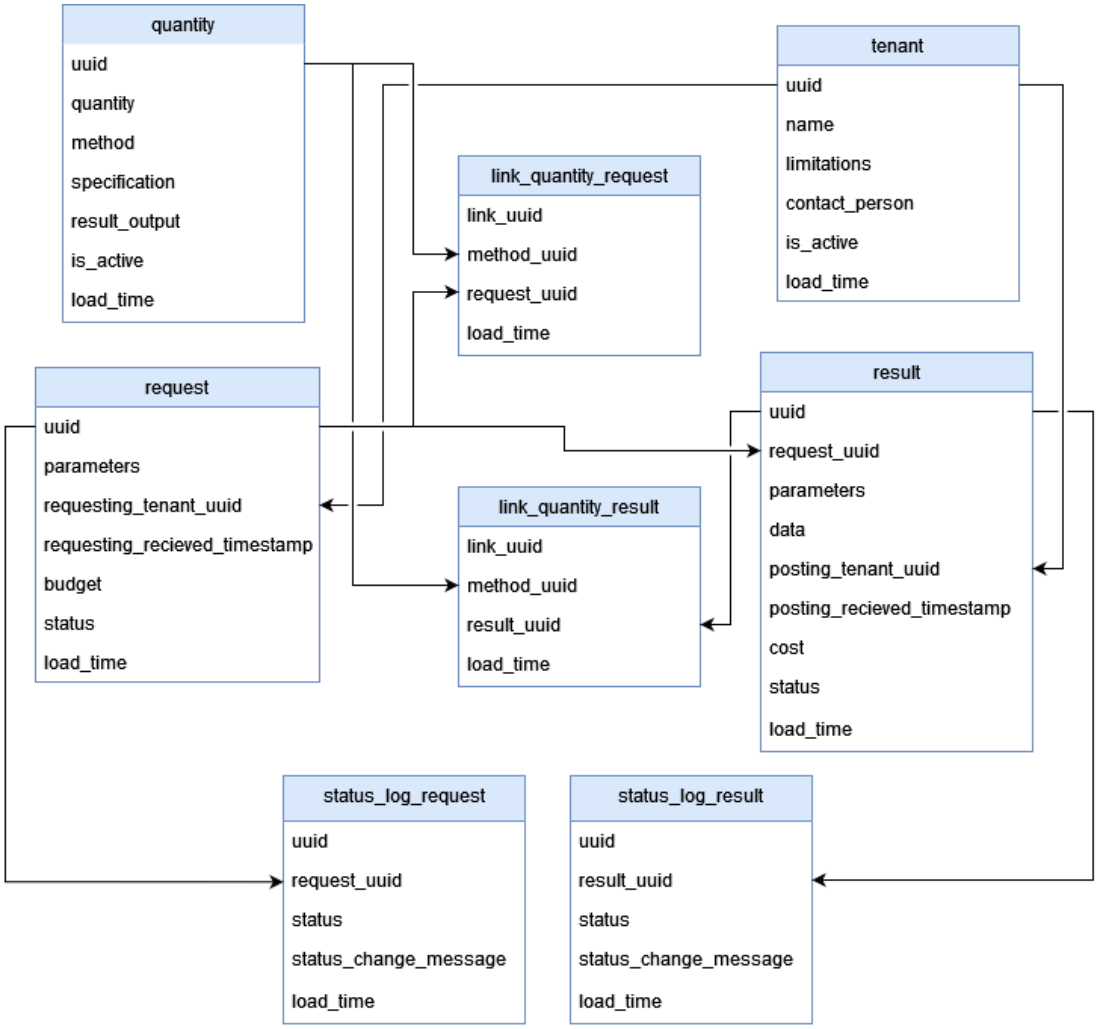


Figure 4: Database schema designed and implemented for FINALES. Arrows indicate table relationships with the direction defining the origin of the primary key mapped to a table as a foreign key. Besides this schema, a separate single table database is used for user credentials.

FINALES is designed to be easily installable in any computer and to be self-contained, so that a new independent installation (with its own independent database and secure access credentials for the tenants) can be used for a given optimisation task, thus preventing any risk of unwanted data leak and keeping different MAP optimisations fully

independent. We call each installation a FINALES *instance*. Once capabilities and tenants are registered in an instance of FINALES, the tenants can communicate to the server through the web API endpoints. The Python framework FastAPI [28] is utilised for constructing the RESTful API. In total, 17 endpoints are exposed to the tenants of which one handles the necessary user authentication protecting the other 16 endpoints. Several of the endpoints are however suited only for functionalities related to an optimiser and archiving tenant. A comprehensive explanation of all the API endpoints can be found in section SI-1.2. The most relevant endpoints for an experimental or computational tenant are `pending_requests/` for retrieving the currently pending requests, `requests/object_id/update_status/` e.g. for changing the status of a request to `reserved` when it has been picked up by a tenant and `results/` for posting results based on a specific request. We stress that the exact parameters in the requests might not be directly applicable to a given tenant implementation. For example, in this study, the electrolyte formulation used in MD simulations may necessitate rounding ratios to achieve an integer number of molecules, resulting in slight deviations from the original formulation. An important design aspect is therefore that the parameters reported in requests and results are allowed to differ, and both are stored in the server.

A core design challenge is the need for the server to provide a rigid framework for brokering information, while remaining flexible to unforeseen circumstances. The proposed solution is to allow the status of requests and results to be changed through endpoints. This accounts for scenarios such as a tenant not being able to fulfil a request, or e.g. leakage of a solution is discovered in an experimental setup after posting results, making the results unreliable. An additional embedded feature allows for prepopulating a database with previously generated results, which enables the deployment of an optimiser with initial training data for the specific optimisation.

4.2 Schemas

The JSON schemas used to define the various quantities and methods involved in this study are available on GitHub with the URL `https://github.com/BIG-MAP/FINALES2_schemas`. Together with the data, most of the tenants also provide metadata, such as a success field reporting whether the method was executed successfully, and a rating providing information about the data quality as judged by the tenant generating the data on a scale from 1 (low quality) to 5 (high quality). An example for a definition of a quality rating is reported in the section SI-1.3 of the SI.

The instances of the optimiser were configured to only consider results with `success == True` and `rating >= 3`, if a rating is available. All entries, which do not fulfil either of these requirements or are marked as `deleted` in their status are considered invalid and therefore disregarded by the optimiser instances.

4.3 Tenants

As mentioned earlier, a tenant is a client connected to FINALES, which may be purely software-based, hardware or even a human researcher performing tasks. The tenants

used in this study are described in the following subsections.

4.3.1 Optimiser tenant (F2Opt)

The FINALES 2 Optimiser (F2Opt) is a software tenant implementing a Bayesian optimisation (BO) procedure. The role and responsibility of F2Opt in the MAP is to request new results from the data-producing methods available in the MAP, with the aim to optimise some predefined objective(s). In general, F2Opt is configured to iteratively consume previously reported results from the FINALES database, use the data to train a machine learning model, and apply a BO procedure to propose new electrolyte formulations which are then included in new requests submitted to FINALES. The specific optimisation task, including what data to consume and which quantities to optimise, is specified in a configuration file and multiple configurations of F2Opt can be deployed in parallel to accommodate the specific objectives of the MAP. In this work, we deployed two instances of F2Opt configured to optimise conductivity (the OCond configuration) and EOL (the OEOL configuration), respectively.

To fully utilise the capabilities of the MAP, the optimiser tenant must be capable of considering measurements of different fidelity and from different data producing methods, including simulations and experiments. In general, data from different sources can have different noise and bias as well as different associated cost of evaluation which need to be considered. Additionally, it can be beneficial to co-optimise multiple objectives at the same time. Therefore, the optimiser tenant should be capable of multi-source multi-objective optimisation to enable a diverse set of optimisation tasks in the MAP. To meet these requirements, the BO procedure implemented in the F2Opt tenant applies a Gaussian process (GP) regression model, which is a smooth and highly flexible model that provides uncertainty estimates and can accommodate small datasets, which is often the starting point in optimisation tasks. These properties make GPs a popular choice of model in BO. To handle data from multiple sources, we consider a multi-task GP [29] and treat each data source as a separate output. This allows for separate predictions for each data source while utilising correlations between data sources to improve the predictions [30]. An acquisition function is then applied to propose new promising electrolyte formulations. In particular the widely used Expected Improvement (EI) function is used. Although not used in this work, multi-objective optimisation can be achieved by transforming multiple objectives into a single objective with a scalarising function [31], which can then be optimised using standard acquisition functions. Additionally, the optimiser can be configured to initially sample a number of random points before switching to the BO procedure when no initial data are available to fit a model. The F2Opt code was developed in Python using the Pandas [32], PyTorch [33] and GPyTorch [34] packages and is available online: `https://github.com/BIG-MAP/F2Opt`.

4.3.2 Molecular Dynamics (MD) tenant

The simulation tenant for calculating lithium ion conductivity is using MD and it is implemented in BIOVIA Pipeline Pilot (BPP) [35]. The tenant is registered in FINALES as

the `3DS_tenant` for the quantity `conductivity` and the method `molecular_dynamics`. A high-level outline of the workflow for the MD simulation tenant is shown in Figure SI-2a in the SI. The workflow for the MD simulation tenant parses the requests from the optimiser tenant to convert the requested electrolyte formulation into an atomistic 3D model of the electrolyte. The molecular structure of each ingredient in the request is described by SMILES and the fractional concentrations are converted into integer numbers of molecules. The restriction to use complete molecules can lead to small differences to the real-valued concentrations requested by the optimiser. The simulation tenant handles up to a few hundred molecules, which places a lower limit on the concentration of each individual ingredient at 0.5 mol.− %. A physical limitation is that the lithium ions start to cluster and become immobile at high salt concentrations, which places an upper limit on the salt concentration around 50 mol.− %. However, this restriction is not significant as we are focused on solutions with high lithium ion mobility. The tenant creates 5 randomly generated amorphous cells for each formulation. These supercells of the type shown in Figure SI-2b in SI are the starting points for a sequence of MD simulations [36] using the COMPASSIII force field [37]. The density is equilibrated in two steps by MD in the NVT ensemble (constant volume, temperature and number of molecules) with the velocity scale thermostat for 200 ps and NPT ensemble (constant pressure, temperature and number of molecules) for 200 ps using the Nose-Hoover-Langevin (NHL) thermostat [38] and the Andersen barostat [39]. The production run is performed by MD in the NVE ensemble (constant volume, total energy and number of molecules) for 2.5 ns. Diffusion coefficients are derived by an analysis of the mean square displacement via the Einstein relation. The conductivity is extracted using the Nernst-Einstein equation [40]. Finally ionic conductivities, alongside the corresponding densities and temperatures for all five configurations, are collected and posted to FINALES. More details about the MD tenant are provided in the SI.

4.3.3 ASAB tenant

The system for the Autonomous Synthesis and Analysis of Battery electrolytes (ASAB) serves as an experimental tenant providing the capability to formulate electrolytes and measure ionic conductivity using a two electrode symmetric electrochemical cell. In FINALES it is registered with the quantities `conductivity` and `electrolyte` with the methods `two_electrode` and `flow`, respectively. The in-house developed electrochemical cell is made up from a polytetrafluoroethylene (PTFE) body equipped with two oppositely positioned stainless steel screws with sanded surfaces serving as the parallel electrodes. The cell was calibrated using the results of repeated measurements of 1 M $LiPF_6$ in EC and EMC (EC:EMC 3:7 by weight) combined with data obtained from the empirical model reported by Ding et al. [18]. Electrochemical Impedance Spectroscopy (EIS) measurements are controlled by a PalmSens4 potentiostat (PalmSens B.V., Houten, Netherlands) and the data are automatically analysed using the MADAP [12] version 1.1.0. The 6 syringe pump modules and 10 eleven-port rotary valve modules (Cetoni GmbH, Korbussen, Germany) integrated into the ASAB hardware are used to formulate electrolytes starting from stock solutions. Since the optimiser requests formulations in

mole fractions, while ASAB doses the stock solutions in volume fractions, the specifications of the formulations need to be transformed. The resulting formulations may only be approximations of the requested one, which is why the ASAB tenant reports the mole fractions obtained from the transformed formulations rounded to two decimals together with the corresponding ionic conductivity results. The hardware of the system is operated inside a nitrogen filled glovebox. Due to a lack of active temperature control inside the glovebox, the temperature is read manually from time to time and entered in a configuration file, from where it is loaded by the ASAB tenant and reported to FINALES together with the ionic conductivity results.

Details regarding the calibration procedure, the hardware setup, and the mode of operation of the ASAB tenant can be found in section SI-2.2.

4.3.4 AutoBASS tenant

The Autonomous Battery Assembly System (AutoBASS) [41] is a modular automatic coin cell assembly system. Devices connected to this system comprise three six-axis robotic arms (Mecademic meca500 rev.3), a precision linear rail (Jenny Science Linax LXS 1800), a digital coin cell crimping machine (MTI MSK-160E, China), two cameras, and a 200 µL dispensing module (Sartorius rLine). The system has the capability to assemble CR2032 coin cells within a nitrogen-filled glovebox and it is registered with FINALES for the quantity `cell_assembly` and the method `autobass_assembly`.

The assembly process is carried out by orchestrated actions performed by the above-mentioned devices, including pick-and-place of cell components, image recognition for the auto-correction of the placement, dispensing of electrolyte, transfer, crimping and placement in the storage holder. Tracking of the process is realised by collecting visual information regarding the placement of components using an integrated camera, which is part of the data generated by AutoBASS. The tenant of AutoBASS is structured based on the reference tenant of FINALES, which is integrated into the interface of the local system. The limitations of the tenant are currently specified in terms of the maximum number of cells and the feasible battery chemistry.

Details regarding the AutoBASS tenant can be found in section SI-2.3.

4.3.5 Cycler tenant

The Cycler tenant and the corresponding data analysis service are build up based on an already existing system of FastAPI servers. The implementation makes use of some features of the manufacturer's software, whereas the analysis is implemented in-house. The tenant is capable of processing reservation requests for cycling channels and handling the tests including the creation of a protocol, start and stop of the test, logging of errors, exporting and saving the data as well as performing selected analyses. To fulfil its tasks, it is registered in FINALES with the quantities `cycling_channel` and `capacity` with the corresponding methods `service` and `cycling`. The cycler saves the status, errors, parameters and the request of each individual channel in a JSON document and consequently is resistant to planned and unexpected stops. The tenant performs

an intermediate export of the capacity trajectories after the duration estimated for at least 40 cycles elapsed. The exported data are subsequently processed by the analysis functionality and posted to FINALES. The experiments continue and are stopped once 200 cycles are finished. A final export is triggered after the estimated time required for 200 cycles elapsed. Since the optimisation of the EOL is based on the predicted EOL, the data obtained from the final export is not posted to FINALES, but stored locally. Details regarding the Cycler tenant can be found in section SI-2.4.

4.3.6 Degradation model tenant

The Degradation model tenant is a purely software-based tenant. It takes data from the first cycles of a cell as the input and outputs a prediction over the capacity trajectory and subsequent EOL along with the associated uncertainty early in the lifetime. It is therefore registered with FINALES as the `EOL_tenant` for the quantity `degradationEOL` and the method `degradation_model`. The underlying machine learning model, a Long Short-Term Memory (LSTM) neural network is based on previous work by Rieger et al. [42]. The tenant receives the capacity trajectory of at least the first 40 cycles along with the overpotential difference between the 10^{th} and 40^{th} cycle, the Coulombic efficiency, and the variance between charge and discharge capacities [43]. Based on this input the model predicts the complete cell trajectory. The EOL is calculated as the point at which the cell capacity goes below 80 % of the initial capacity. To capture uncertainty, we use an ensemble of five models and extend the LSTM architecture to predict the variance of the output along with the mean. If the cell chemistry and format from which training data are obtained differ from the production data, the model predictions can become unreliable. Mitigating this risk, the model is trained with a dataset of 44 cells obtained from reference [44] after filtering out outliers. This data was generated using the same cell type, hardware, and cycling protocol as the Cycler tenant. The uncertainty output gives an indication of the prediction accuracy. The code for this tenant was developed in PyTorch [33] and is available online: `https://github.com/BIG-MAP/eol_degradation_tenant`.

4.3.7 Transportation tenant

The Transportation tenant deals with the transport of physical samples between devices. In this study, electrolytes and coin cells need to be transported between different setups. The transport itself is done manually, but the Transportation tenant handles the communication with FINALES. It is set up as a chat service and registered in the FINALES database with the quantity `transport` and the method `transport_service`. Once it picks up a request from FINALES, it broadcasts the information regarding the physical location of the sample and the requested destination to the clients registered with the internal chat service. The humans monitoring the chat perform the transport and confirm it by a short, predefined chat message, which triggers the tenant to post a successful response to FINALES.

4.3.8 Overlort

The Overlort is a purely software-based workflow tenant to enable requests for predicted EOL. In this study, the Overlort connects the hardware tenants of ASAB, AutoBASS and the Cycler, as well as the software-based Transportation and Degradation model tenants in a workflow. In FINALES, it is associated with the quantity `degradationEOL` and the method `degradation_workflow`. The limitations of the ASAB, AutoBASS, Cycler and Degradation model tenants determine its accessible parameter space. Details regarding the Overlort tenant can be found in section SI-2.5.

4.3.9 Archiving tenant

The Archiving tenant is a command-line client capable of fetching data from FINALES and submitting the retrieved data to the API offered by the BIG-MAP Archive data repository [15]. Since this tenant does not report results to FINALES and acts in the background, it is not registered as a tenant in the FINALES database, but is granted access to the data by specific credentials. It serves two specific purposes: (i) preventing data loss in the event of potential corruption of the FINALES database and (ii) promptly disseminating results of calculations and experiments shortly after their submission to the server, to a broader consortium of researchers (in this case, the BIG-MAP project members). The archiving is executed periodically by means of a "cron" job scheduler. The tenant is optimised for efficient use of storage space and minimal upload time, by linking unchanged files between versions of the same entry, and only creating one new entry per campaign. Details regarding the Archiving tenant can be found in section SI-2.6. Overall the defined data structures and archiving tenant (including the ontological mapping) conform FINALES with the data management plan developed for the BIG-MAP project [45] .

Ontology The communication among tenants relies on the compliance to the commonly agreed exchange syntax underlying the structure of the requests and results posted for each method. As the number of tenants grows and their functions diversify, the data they exchange becomes structurally complex and susceptible to ambiguities. JSON keys might repeat but refer to different concepts depending on their location in the schema. E.g., *temperature* within a conductivity measurement might mean the internal temperature of an electrolyte solution, surface temperature of the cell, ambient temperature for the chamber, etc. In addition to ambiguity, the data exchanged by tenants might not be readily accessible. The exchange schemas are designed to support communication between tenants, but such flexibility comes at the expense of syntactic complexity. For instance, retrieving a conductivity result requires knowing the full data structure (albeit this can be retrieved and investigated directly from FINALES) to locate the relevant fields and values.

A remedy to ambiguity and accessibility is mapping data to controlled vocabularies of concepts. Such mapping effectively upgrades a syntactic data model into a semantic data model, where data fields are attributed an unambiguous meaning. As a *tempera-*

ture key in the schema is mapped to the concept of *SurfaceTemperature* in a controlled vocabulary, the intended meaning of the field becomes explicit and distinct to, e.g., *SampleTemperature*. Moreover, the association between a measurement (uniquely identified with a UUID), a vocabulary concept (e.g. Schema.org concept `dateCreated`), and a value (e.g. `2024-02-16`) can be represented as a Node-Edge-Node connection in a network. Representing results as these *Triples* [46] in a Graph Database makes data accessible irrespective of how it is syntactically encoded in the data model.

Identifiers should ideally be unique and resolve to a web resource that offers more information about the entity. For instance, the concept of Electrolyte is uniquely identified with an Internationalised Resource Identifier (IRI) [47] that resolves in a web browser to a description of the concept: `https://w3id.org/emmo/domain/electrochemistry#electrochemistry_fb0d9eef_92af_4628_8814_e065ca255d59`. If the identifier cannot be resolved, we attempt using URL prefixes that resolve to some general information about the object. The implementation will facilitate search in a knowledge base. The ultimate purpose of this implementation is to make FINALES results available in a graph database that can be flexibly queried using structured languages, such as SPARQL [48] or CYPHER [49].

To enrich the data with semantically well-defined concepts, we employ the recently developed ontology Battery Interface Ontology (BattINFO). BattINFO describes thousands of concepts related to electrochemistry and batteries. Concepts are linked to each other adhering to logically consistent relationships, thus rendering meaning machine-readable [13]. The connections among data and concepts must be encoded using a standard method. In this work, JSON for Linked Data (JSON-LD) is used to link the results from the conductivity and EOL tasks to ontology concepts in BattINFO and Schema.org. BattINFO describes measurement fields related to batteries and electrochemistry, while the Schema.org vocabulary describes tenant metadata, software resources and other non-electrochemical concepts. For example, we enrich the descriptions of the request/result posting tenant with the ORCID of the responsible person linked to the schema:author type, and ROR-ID for the organisation linked to schema:creator type. These choices improve data consistency, since they avoid reliance on other identifiers (such as email addresses) that are subject to change. EOL and conductivity results in this study are all expressed in JSON-LD files mapped to ontology concepts. Details and an example of the semantic implementation of experimental results can be found in section SI-2.6 of the SI. Development of the knowledge graph representation of the data was done subsequently to the data collection. The graphs were therefore uploaded to the archive entries after the optimisation phases had been completed.

4.4 The structure of this study

The campaign presented here comprises two optimisation tasks with separate objectives. The first task is to identify an electrolyte formulation with maximum ionic conductivity by using data generated from experiments and simulations. As for the second task, the objective is to find an electrolyte formulation that maximises the EOL of coin cells. Each of these tasks is associated with a separate instance and configuration of the F2Opt

tenant, OCond and OEOL. We show, that the two optimisers can operate on the same MAP through a common FINALES instance, while pursuing fully independent tasks.

The tenants involved in the *single-task phase* are the optimiser (OCond), the MD tenant, the ASAB tenant, and the Archiving tenant. This phase is also used to confirm the limitations set for these tenants or adjust them, if needed. In the *multi-task* phase, a request with the conductivity-optimised formulation, as well as two non-optimised formulations identified from the results obtained in the single-task phase of the campaign are manually submitted to FINALES to investigate potential differences in the cycling performance. The electrolyte system chosen for this study is composed of $LiPF_6$ dissolved in EC and EMC, because this is a well-investigated electrolyte system for which references are available [12, 18, 20, 23–25].

4.4.1 Single-task phase — ionic conductivity only

The single-task phase of the campaign was run from the 26$^{\text{th}}$ of September 2023 to the 28$^{\text{th}}$ of October 2023. It included the ASAB tenant and MD tenant, which posted its first result 8 days after the first result was posted by the ASAB tenant.

After the end of the single-task phase, a total of 81 entries are stored in the FINALES database. 12 of these entries are invalid. 72 of the total entries have a quality rating of 4 or 5. Only 3 entries have a rating of 3, while 6 have a rating of 2. In total, 69 entries in the database are valid according to the requirements regarding the status and the rating configured in the OCond optimiser.

4.4.2 Multi-task phase — ionic conductivity and end-of-life

The multi-task phase of the campaign included the optimisation of the EOL and the ionic conductivity in parallel. The MD tenant and the ASAB tenant started their contribution of results to the database approximately 5 hours apart, which is significantly shorter than 8 days in the single-task phase. Both optimiser instances, OCond and OEOL, were running at the same time, concurrently posting requests for their respective task. The multi-task phase was started on the 6$^{\text{th}}$ of November 2023 and lasted until the last result was posted on the 14$^{\text{th}}$ of December 2023. During this time, a total of 274 result entries were posted to FINALES. In the multi-task phase, 7 quantities and 9 methods were registered with FINALES. Out of the 274 results, 173 are valid.

5 Data availability

The data generated in the course of this study is publicly available under the DOI `https://doi.org/10.24435/materialscloud:qt-1s`

6 Code availability

The code for the version 1.1.0 of the FINALES framework is available under the DOI `https://doi.org/10.5281/zenodo.10987727` and on GitHub `https://github.com/`

BIG-MAP/FINALES2. The code of the FINALES schemas version 1.0.1 is available under the DOI https://zenodo.org/records/11142866and the latest version of FINALES schemas can be found on GitHub https://github.com/BIG-MAP/FINALES2_schemas. The code associated with the individual tenants is available as stated in Table 1:

Table 1: The code availability for each of the tenants presented in this study.

Tenant	**Version**	**Code availability**
F2Opt	v1.0.0	https://github.com/BIG-MAP/F2Opt https://github.com/BIG-MAP/F2Opt/releases/tag/v1.0.0
ASAB	v2.0.1	https://github.com/Helge-Stein-Group/ASAB https://doi.org/10.5281/zenodo.11146699
ASAB tenant	v1.0.0, v1.0.1	https://github.com/BIG-MAP/FINALES_ASAB_tenant https://doi.org/10.5281/zenodo.11144341
Overlort	v1.0.0	https://github.com/BIG-MAP/FINALES_Overlort_tenant https://doi.org/10.5281/zenodo.11145783
AutoBASS	v1.0.1	https://github.com/BIG-MAP/FINALES_AutoBASS_tenant https://doi.org/10.5281/zenodo.11145983
Cycler	v1.0.0	https://github.com/BIG-MAP/FINALES_Cycler_tenant https://doi.org/10.5281/zenodo.11145850
Degradation model	v1.0.0	https://github.com/BIG-MAP/eol_degradation_tenant https://github.com/BIG-MAP/eol_degradation_tenant/releases/tag/v1.0.0
Archiving	v1.1.0	https://github.com/materialscloud-org/big-map-archive-api-client/blob/2511f86ec08c44fc8d32e5d96953eb72e3ac89d9/README.md#back-up-finales-databases https://zenodo.org/records/11184798
Transportation	v1.0.0	https://github.com/BIG-MAP/FINALES_Transportation_tenant https://doi.org/10.5281/zenodo.11120059

7 Acknowledgements

This project received funding from the European Union's Horizon 2020 research and innovation program under grant agreement no. 957189 (BIG-MAP). The authors acknowledge BATTERY2030PLUS, funded by the European Union's Horizon 2020 research and innovation program under grant agreement no. 957213. This work contributes to the research performed at CELEST (Center for Electrochemical Energy Storage Ulm-Karlsruhe) and was co-funded by the German Research Foundation (DFG) under Project ID 390874152 (POLiS Cluster of Excellence). HSS acknowledges funding from DFG EXC 2089/1-390776260 (e-conversion). TV acknowledges funding from the Pioneer Center for Accelerating Materials Discovery (CAPeX), DNRF Grant P3. FFR, FL and GP acknowledge funding by the NCCR MARVEL, a National Centre of Competence in Research, funded by the Swiss National Science Foundation (grant number 205602). GP acknowledges funding by the Open Research Data Program of the ETH Board (project "PREMISE": Open and Reproducible Materials Science Research). FL and GP acknowledge useful discussions and support by Valeria Granata in the setup of the Archiving tenant. MV acknowledges fruitful discussions with Jackson K. Flowers regarding the transformation of formulations in the ASAB tenant. LM and MV acknowledge effective discussions with Christian Wölke concerning the cycling data obtained from the pretests.

8 Author contributions

Conceptualization: AB, HSS; Data curation: JB, SKS, EF, MV, FL; Formal analysis: JB, MV, JC, HH, FH, LM, AB; Funding acquisition: IEC, GP, JC, HH, FH, HSS, TV, AB; Investigation: JB, MV, LM, BZ, SKS; Methodology: JB, SKS, MV, FFR, JC, HH, FH, MG, LHR, AS, SF, SC, HSS, AB; Project administration: SKS, MV; Resources: BZ; Software: JB, SKS, EF, MV, FFR, FL, GP, JC, HH, FH, LM, LHR, BZ, SC, HSS; Supervision: IEC, GP, JC, HH, FH, HSS, TV, AB; Validation: JB, MV, MG, AS, SF; Visualization: JB, SKS, MV, JC, HH, FH; Writing - original draft: JB, SKS, EF, MV, FL, JC, HH, FH, LHR, HSS, TV, AB; Writing - review & editing: JB, IEC, SKS, EF, MV, FFR, GP, JC, HH, FH, MG, LM, LHR, BZ, AS, SF, SC, HSS, TV, AB

References

[1] Milad Abolhasani and Eugenia Kumacheva. "The rise of self-driving labs in chemical and materials sciences". In: *Nature Synthesis* 2.6 (2023), pp. 483–492.

[2] Martin Seifrid et al. "Reaching critical MASS: crowdsourcing designs for the next generation of materials acceleration platforms". In: *Matter* 5.7 (2022), pp. 1972–1976.

[3] Monika Vogler et al. "Brokering between tenants for an international materials acceleration platform". In: *Matter* 6.9 (2023), pp. 2647–2665. ISSN: 2590-2385. DOI: https://doi.org/10.1016/j.matt.2023.07.016. URL: https://www.sciencedirect.com/science/article/pii/S2590238523003739.

[4] Arghya Bhowmik et al. "A perspective on inverse design of battery interphases using multi-scale modelling, experiments and generative deep learning". In: *Energy Storage Materials* 21 (2019), pp. 446–456.

[5] Williams Agyei Appiah et al. "Sensitivity analysis methodology for battery degradation models". In: *Electrochimica Acta* 439 (2023), p. 141430.

[6] Felix Strieth-Kalthoff et al. "Delocalized, Asynchronous, Closed-Loop Discovery of Organic Laser Emitters". In: (2023).

[7] Ilyes Batatia et al. *A foundation model for atomistic materials chemistry*. 2024. arXiv: 2401.00096 [physics.chem-ph].

[8] Julia Amici et al. "A roadmap for transforming research to invent the batteries of the future designed within the european large scale research initiative battery 2030+". In: *Advanced energy materials* 12.17 (2022), p. 2102785.

[9] Nathan J Szymanski et al. "An autonomous laboratory for the accelerated synthesis of novel materials". In: *Nature* 624.7990 (2023), pp. 86–91.

[10] Chi Chen et al. "Accelerating computational materials discovery with artificial intelligence and cloud high-performance computing: from large-scale screening to experimental validation". In: *arXiv preprint arXiv:2401.04070* (2024).

[11] Laura Hannemose Rieger et al. "Understanding the patterns that neural networks learn from chemical spectra". In: *Digital Discovery* 2 (6 2023), pp. 1957–1968.

[12] Fuzahn Rahmanian. *Modular and Autonomous Data Analysis Platform (MADAP)*. Version v1.1.0. Aug. 7, 2023. DOI: 10.5281/ZENODO.7374382. URL: https://zenodo.org/record/7374382 (visited on 12/03/2023).

[13] Simon Clark et al. "Toward a unified description of battery data". In: *Advanced Energy Materials* 12.17 (2022), p. 2102702.

[14] Simon Clark et al. *BIG-MAP/BattINFO: v0.6.0*. Version v0.6.0. Aug. 2023. DOI: 10.5281/zenodo.8260800. URL: https://doi.org/10.5281/zenodo.8260800.

[15] François Liot et al. *BIG-MAP Archive*. Feb. 2023. URL: https://archive.big-map.eu/.

[16] Leopold Talirz et al. "Materials Cloud, a platform for open computational science". In: *Sci. Data* 7 (1 2020), p. 299. DOI: doi.org/10.1038/s41597-020-00637-5.

[17] Michael S. Ding, Kang Xu, and T. Richard Jow. "Liquid-Solid Phase Diagrams of Binary Carbonates for Lithium Batteries". In: *Journal of The Electrochemical Society* 147.5 (2000), p. 1688. DOI: 10.1149/1.1393419. URL: https://dx.doi.org/10.1149/1.1393419.

[18] M. S. Ding et al. "Change of Conductivity with Salt Content, Solvent Composition, and Temperature for Electrolytes of LiPF[sub 6] in Ethylene Carbonate-Ethyl Methyl Carbonate". In: *Journal of The Electrochemical Society* 148.10 (2001), A1196. ISSN: 00134651. DOI: 10.1149/1.1403730. URL: https://iopscience.iop.org/article/10.1149/1.1403730 (visited on 12/03/2023).

[19] Fuzhan Rahmanian et al. "One-Shot Active Learning for Globally Optimal Battery Electrolyte Conductivity**". In: *Batteries & Supercaps* 5.10 (2022), e202200228. DOI: https://doi.org/10.1002/batt.202200228. eprint: https://chemistry-europe.onlinelibrary.wiley.com/doi/pdf/10.1002/batt.202200228. URL: https://chemistry-europe.onlinelibrary.wiley.com/doi/abs/10.1002/batt.202200228.

[20] Anand Narayanan Krishnamoorthy et al. "Data-Driven Analysis of High-Throughput Experiments on Liquid Battery Electrolyte Formulations: Unraveling the Impact of Composition on Conductivity**". In: *Chemistry–Methods* 2.9 (2022), e202200008. DOI: https://doi.org/10.1002/cmtd.202200008. eprint: https://chemistry-europe.onlinelibrary.wiley.com/doi/pdf/10.1002/cmtd.202200008. URL: https://chemistry-europe.onlinelibrary.wiley.com/doi/abs/10.1002/cmtd.202200008.

[21] Johannes Landesfeind and Hubert A. Gasteiger. "Temperature and Concentration Dependence of the Ionic Transport Properties of Lithium-Ion Battery Electrolytes". In: *Journal of The Electrochemical Society* 166.14 (Sept. 2019), A3079. DOI: 10.1149/2.0571912jes. URL: https://dx.doi.org/10.1149/2.0571912jes.

[22] Eibar Flores et al. "Learning the laws of lithium-ion transport in electrolytes using symbolic regression". In: *Digital Discovery* 1 (4 2022), pp. 440–447. DOI: 10.1039/D2DD00027J. URL: http://dx.doi.org/10.1039/D2DD00027J.

[23] Behnam Ghalami Choobar et al. "Multiscale Investigation on Electrolyte Systems of [(Solvent + Additive) + LiPF6] for Application in Lithium-Ion Batteries". In: *The Journal of Physical Chemistry C* 123.36 (2019), pp. 21913–21930. DOI: 10.1021/acs.jpcc.9b04786. eprint: https://doi.org/10.1021/acs.jpcc.9b04786. URL: https://doi.org/10.1021/acs.jpcc.9b04786.

[24] Andreas Nyman, Mårten Behm, and Göran Lindbergh. "Electrochemical characterisation and modelling of the mass transport phenomena in LiPF6–EC–EMC electrolyte". In: *Electrochimica Acta* 53.22 (2008), pp. 6356–6365. ISSN: 0013-4686. DOI: https://doi.org/10.1016/j.electacta.2008.04.023. URL: https://www.sciencedirect.com/science/article/pii/S0013468608005045.

[25] E. R. Logan et al. "A Study of the Transport Properties of Ethylene Carbonate-Free Li Electrolytes". In: *Journal of The Electrochemical Society* 165.3 (Mar. 2018), A705. DOI: 10.1149/2.0981803jes. URL: https://dx.doi.org/10.1149/2.0981803jes.

[26] Richard D Hipp. *SQLite*. Version 3.43. 2023. URL: https://www.sqlite.org/index.html.

[27] Michael Bayer. "SQLAlchemy". In: *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*. Ed. by Amy Brown and Greg Wilson. aosabook.org, 2012. URL: http://aosabook.org/en/sqlalchemy.html.

[28] S. Ramírez. *FASTAPI*. 2023. URL: https://fastapi.tiangolo.com/.

[29] Edwin V Bonilla, Kian Chai, and Christopher Williams. "Multi-task Gaussian Process Prediction". In: *Advances in Neural Information Processing Systems*. Ed. by J. Platt et al. Vol. 20. Curran Associates, Inc., 2007. URL: https://proceedings.neurips.cc/paper_files/paper/2007/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf.

[30] Kevin Swersky, Jasper Snoek, and Ryan P Adams. "Multi-Task Bayesian Optimization". In: *Advances in Neural Information Processing Systems*. Ed. by C.J. Burges et al. Vol. 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper_files/paper/2013/file/f33ba15effa5c10e873bf3842afb46a6-Paper.pdf.

[31] Tinkle Chugh. "Scalarizing Functions in Bayesian Multiobjective Optimization". In: *2020 IEEE Congress on Evolutionary Computation (CEC)*. 2020, pp. 1–8. DOI: 10.1109/CEC48606.2020.9185706.

[32] Wes McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.

[33] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[34] Jacob R Gardner et al. "GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration". In: *Advances in Neural Information Processing Systems*. 2018.

[35] Dassault Systèmes Americas Corporation. *BIOVIA Pipeline Pilot*. Version Release 2023. Dec. 2022. URL: https://www.3ds.com/products/biovia/pipeline-pilot.

[36] Dassault Systèmes Americas Corporation. *BIOVIA Materials Studio*. Version Release 2023. Dec. 2022. URL: https://www.3ds.com/products/biovia/materials-studio.

[37] Reinier LC Akkermans, Neil A Spenley, and Struan H Robertson. "COMPASS III: Automated fitting workflows and extension to ionic liquids". In: *Molecular Simulation* 47.7 (2021), pp. 540–551. DOI: https://doi.org/10.1080/08927022.2020.1808215.

[38] Mark A. J. Chaplain Alex A. Samoletov Carl P. Dettmann. "Thermostats for "slow" configurational modes". In: *J. Stat. Phys.* 128 (2007), pp. 1321–1336. DOI: https://doi.org/10.1007/s10955-007-9365-2).

[39] H. C. Andersen. "Molecular dynamics simulations at constant pressure and/or temperature". In: *J. Chem. Phys.* 72 (1980), p. 2384. DOI: https://doi.org/10.1063/1.439486.

[40] "Multi-Scale Electrolyte Transport Simulations for Lithium Ion Batteries". In: *J. Electrochem. Soc.* 167.9 (2020), p. 013522. DOI: DOI:10.1149/2.0222001JES. URL: https://iopscience.iop.org/article/10.1149/2.0222001JES.

[41] Bojing Zhang et al. "Robotic cell assembly to accelerate battery research". In: *Digital Discovery* 1 (6 2022), pp. 755–762. DOI: 10.1039/D2DD00046F. URL: http://dx.doi.org/10.1039/D2DD00046F.

[42] Laura Hannemose Rieger et al. "Uncertainty-aware and explainable machine learning for early prediction of battery degradation trajectory". In: *Digital Discovery* 2.1 (2023), pp. 112–122.

[43] Kristen A Severson et al. "Data-driven prediction of battery cycle life before capacity degradation". In: *Nature Energy* 4.5 (2019), pp. 383–391.

[44] Bojing Zhang et al. *Cycling Data of 64 Cells manufactured by AutoBASS*. 2022. DOI: 10.5281/ZENODO.7299473.

[45] Ivano E. Castelli et al. "Data Management Plans: the Importance of Data Management in the BIG-MAP Project". In: *Batter. Supercaps* 4 (2021), pp. 1803–1812. ISSN: 12. DOI: https://doi.org/10.1002/batt.202100117. URL: https://chemistry-europe.onlinelibrary.wiley.com/doi/10.1002/batt.202100117.

[46] Shelley Powers. *Practical RDF: solving problems with the resource description framework.* " O'Reilly Media, Inc.", 2003.

[47] Martin Duerst and Michel Suignard. *Rfc 3987: Internationalized resource identifiers (iris)*. 2005.

[48] Renzo Angles et al. "Foundations of modern query languages for graph databases". In: *ACM Computing Surveys (CSUR)* 50.5 (2017), pp. 1–40.

[49] Nadime Francis et al. "Cypher: An evolving query language for property graphs". In: *Proceedings of the 2018 international conference on management of data*. 2018, pp. 1433–1445.