

Experiment Specification, Capture and Laboratory Automation Technology (ESCALATE): a software pipeline for automated chemical experimentation and data management

Ian M. Pendleton  and **Gary Cattabriga**, Department of Chemistry, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania 19041, USA
Zhi Li, Molecular Foundry, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, California 94720, USA
Mansoor Ani Najeeb, Department of Chemistry, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania 19041, USA
Sorelle A. Friedler, Department of Computer Science, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania 19041, USA
Alexander J. Norquist , Department of Chemistry, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania 19041, USA
Emory M. Chan , Molecular Foundry, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, California 94720, USA
Joshua Schrier , Department of Chemistry, Fordham University, 441 E. Fordham Road, The Bronx, New York, 10458, USA

Address all correspondence to Joshua Schrier at jschrier@fordham.edu

(Received 15 January 2019; accepted 22 May 2019)

Abstract

Applying artificial intelligence to materials research requires abundant curated experimental data and the ability for algorithms to request new experiments. ESCALATE (Experiment Specification, Capture and Laboratory Automation Technology)—an ontological framework and open-source software package—solves this problem by providing an abstraction layer for human- and machine-readable experiment specification, comprehensive and extensible (meta-) data capture, and structured data reporting. ESCALATE simplifies the initial data collection process, and its reporting and experiment generation mechanisms simplify machine learning integration. An initial ESCALATE implementation for metal halide perovskite crystallization was used to perform 55 rounds of algorithmically-controlled experiment plans, capturing 4336 individual experiments.

Introduction

Chemistry and materials science are entering a new data-driven age,^[1–3] in which planning algorithms select experiments to be conducted by humans or performed autonomously using laboratory robotics.^[4–6] Laboratory automation has been an ongoing endeavor for nearly a quarter century with seminal demonstrations of high-throughput materials research performed by Xiang et al. in 1995.^[7] Subsequent research, predominantly in combinatorial chemistry, focused on the development of high-throughput techniques targeting new material syntheses^[8] and methods of characterization^[9] that have been the topic of several comprehensive reviews.^[10–17] Recent advances in machine learning and artificial intelligence allow for extracting further physical insights latent within these results.^[18,19] Important themes include comprehensive capture and analysis of “successful” and “failed” experiments,^[20–23] adaptively modifying the experiment plans as data are collected,^[4,24–29] machine-learned characterization of experimental outcomes,^[30] ensuring sufficient sampling of relevant experimental variables by avoiding human biases,^[24,31] and integration with chemical informatics features^[32] and physical simulation outputs as machine learning model inputs.^[33–35]

The types of chemistry, experimental processes, level of automation, and throughput of modern materials research

vary tremendously. Representative examples of the current state-of-the-art range from real-time control of single batches of carbon nanotubes^[27] and continuous flow organic synthesis,^[26] to tens of hydrothermal syntheses performed with the help of humans,^[20,23,24] to hundreds or thousands of reactions performed in microwell plates^[21] and in droplets,^[22,25] up to national-level synchrotron facilities.^[36]

The software implemented to generate experiments and collect data for machine learning are similarly diverse. Software packages for materials chemistry have focused on closed-loop operation with specific experiments, robotic hardware, and algorithms, such as ARES to study single-walled carbon nanotubes,^[27] NREL-HTEM for physical vapor deposition of inorganic thin films,^[37] AIR-Chem targeting inorganic perovskite quantum dots,^[38] as well as a number of solutions for polymer chemistry.^[39,40] Like these, ChemOS, a recently released modular software environment for autonomous laboratory operation, focuses primarily on optimization rather than comprehensive data capture.^[41] (We note in passing software libraries for managing closed-loop *computational* organic,^[42] materials,^[15,43] and heterogeneous catalyst^[44] discovery workflows.)

Generalizable automation and data capture in biology is more mature and includes variants such as Wet Lab

Accelerator,^[45] Autoprotocol,^[46] laboratory automation such as Par-Par^[47] and Roboliq,^[48] full experimental workflow and design with integrated LIMS such as Aquarium,^[49] and services offered by companies such as Emerald Cloud Lab^[50] and Transcrip^[51,52]

Despite the advances in related fields described above, these technologies have experienced limited integration into the chemistry and materials science communities. Furthermore, current technology environments provide limited support for human–robot interfacing, do not support the organization of unstructured or unprocessed data, and often are developed without intention for distribution of curated datasets to non-domain expert computer scientists (i.e., machine learning experts). The development of an extensible framework for materials science data collection should thereby include the following components: First, a generalizable mechanism for *specifying* machine-readable experiment plans that enable operations via a web-based application programming interaction (API). Second, the facilitation of hybrid human–robot laboratory operations that enable the recording of data and optimization of experiments. Third, a comprehensive *capture* of data and metadata for a complete description of an experiment, involving both human operations and associated machine-generated files. Finally, processing the data into *reports* that facilitate subsequent use in machine learning algorithms operated by both domain expert chemists and non-domain expert computer scientists.

We describe here a general data model and associated software pipeline code—ESCALATE (Experiment Specification, Capture and Laboratory Automation Technology)—that addresses each of the needs discussed above. As an illustrative specific example, we describe the ESCALATE implementation for high-throughput exploratory synthesis of single-crystal metal halide perovskites, a class of materials that has attracted a great deal of both fundamental^[53–55] and technologic interest for application in photovoltaics,^[56,57] sensors,^[58] lighting,^[59,60] and batteries.^[61] The experimental hardware, chemical process details, and scientific outcomes of this “robot-ready perovskite” synthesis method will be described elsewhere. This example is used to showcase how the ESCALATE data model addresses challenges related to capturing the distinction between experimental intent and reality, hybrid human–machine laboratory operations, data capture for multi-step multi-component reagent preparation, handling increasingly complex datasets for evolving chemistry workflows, and curating datasets useable by non-domain experts for machine learning.

Lifecycle of a metal halide perovskite experiment

A robot-ready metal halide perovskite synthesis provides a concrete example of our ESCALATE framework, and illustrates a number of challenges associated with experiment specification and capture that are solved by ESCALATE. Reagent stock solutions consisting of one or more organic, inorganic, and solvent chemicals must be prepared by a human operator, in a

process that includes manual weighing, followed by heating and mixing. The intended compositions of these reagent stock solutions cannot exceed the solubility limits of the species. The resulting reagent solutions are then moved to a robotic liquid handler capable of pipetting specified volumes into a set of 96 individually-addressed glass vials (denoted as a “*well*”). In addition to the order and timing of the reagent additions specific to each individual *well*, time-varying heating and shaking conditions can be applied globally to the entire collection of vials (denoted as a “*plate*”). The total volume of all species in a valid experiment cannot exceed the volume limits of the vial in which the experiment is contained. At the conclusion of the synthesis, each vial is photographed from several angles, and scored for the formation of crystals. Each well is an individual experimental entity, described by its own template, model, and object information, but sharing common descriptions of reagent stock solutions and plate-wide conditions.

The lifecycle of an experiment starts as a general “*template*”, which is a structured, fill-in-the-blank form that provides a general pattern of information needed to specify an experiment and any relevant experimental limits (see the left side of Fig. 1). Specifying the prerequisite portions of the experimental template generates a “*model*,” which contains the intended laboratory actions and materials. Executing the notional plan described by a *model* results in an “*object*” representing the particular physical laboratory instantiation. Each object is characterized by its particular *type*, *material*, *action*, *observation*, and *outcome* data, see Table I, as well as the intended plan (model) and general experimental constraints (template). Treating entities in this fashion allows us to track the experimental intent, reproduce the particular experimental execution, and clarify the relationship between the nominal experiment and the empirical observations.

Each experimental entity proceeds through four states, as shown in the right side of Fig. 1. The entity begins as a set of templates describing the general process and experimental constraints that must be satisfied. Chemicals are specified as raw ingredients, described in terms of standardized identifiers (e.g., InChIKey) with a set of known attributes (e.g., molecular formula, molecular weight, density). A reagent-type (R) template asks the user to specify specific chemicals and preparation parameters (e.g., mixing time and temperature) and enforces composition constraints (e.g., solubility constraints as a maximum concentration). An experiment-type (E) template asks the user to specify both well- (e.g., statistical distributions of the different reagent volumes to use) and plate-level intent (e.g., addition times, temperature, and mixing conditions) and enforces experimental constraints (e.g., minimum and maximum well-volume limits and operating temperatures). Encoding the experimental constraints into the template allows this to be generalized to other experiment types.

The operator provides the required inputs to the templates using an ESCALATE executable to formally start the generation of models. ESCALATE returns the generated models to the operator. Reagent models include nominal preparation

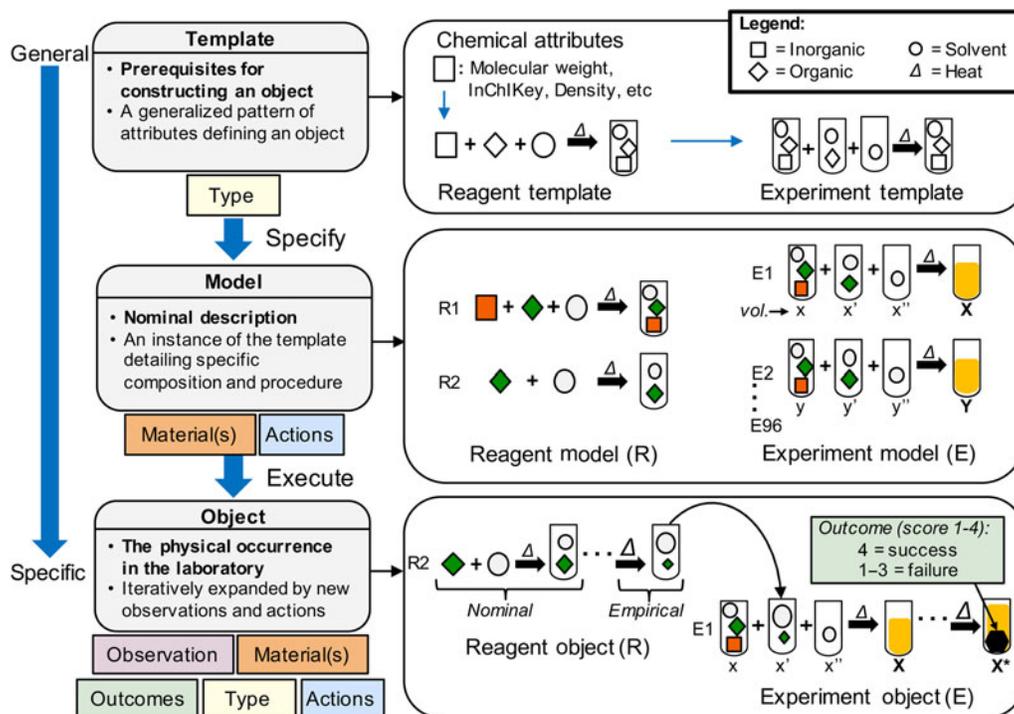


Figure 1. Lifecycle of a metal halide perovskite experimental object starting from the specification of a template experiment through the execution of each experiment object.

instructions including *actions*, and *materials* necessary to prepare the model. Experiment models contain associations with the relevant reagent models as well as the necessary information to perform the experiment in the laboratory.

“*Objects*” describe the physical reagents and experiments conducted in the laboratory according to the instructions in the model. Data collected about objects are distributed across the *type*, *material*, *action*, *observation*, and *outcome* categories as the experiment proceeds (see Table I). For metal halide perovskite experiments, the empirical observations of reagents and experiments include data such as solution volumes, measured temperatures, and concentrations. ESCALATE provides forms for the operator to record observations regarding reagent and experiment objects. The operator aims to capture variance

between the nominal specification of the model and the empirical observations associated with a given object. These data are especially important for accurately capturing the properties which exhibit non-ideality during object preparation (i.e., concentration variance resulting from non-ideal solution mixtures).

For metal halide perovskite experiments, the operator records two data points in the final step: (i) scoring based on optical microscopy and (ii) photographic images of each well of the robotic plate (Fig. 2). Optical microscopy data are recorded as *outcomes* on a scale from 1 to 4, where 1–3 indicates a “failure” to produce large single significant crystals, and 4 indicates the “success” of producing large single crystals.

The “*type*” category describes general information about the nature of the entity, who interacted with the entity, where the

Table I. Chemical ontology for data collection outlining categories of data collection throughout the pipeline.

Category	Overview	Examples
Type	Metadata template	Location, date, time, additional associated types
Material	Experiment inputs	Chemical identity, reagents, equipment, instruments
Action	Materials process	Heating/cooling (time temperature), stirring, reaction time
Observation	Empirical data	Chemical measurements, pH, empirical volumes
Outcome	Product(s) of the experiment	Optical microscopy(classification), reaction images, XRD, NMR

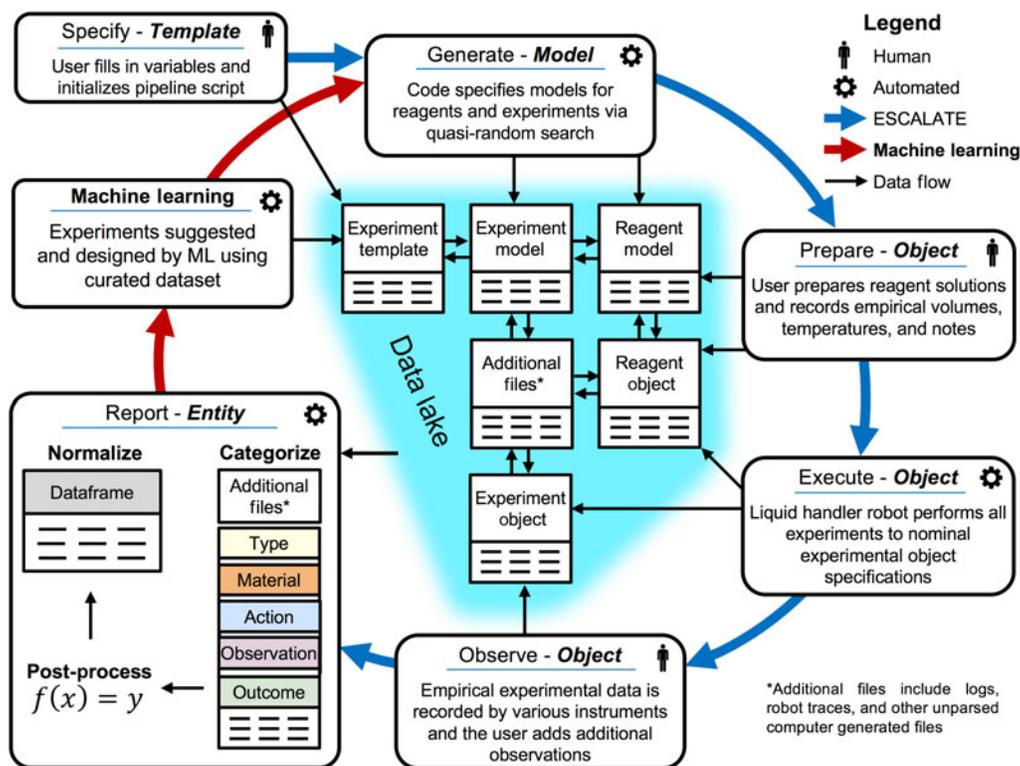


Figure 2. Operational overview of ESCALATE detailing data collection forms and distribution of data into ontologic components leading to a final curated data frame.

entity was created, as well as any associated entities and relevant relationships. Data grouped by *type* record metadata relating to an entity or to group similar entities together. The “*material*” category describes the physical components that are part of the entity, such as chemicals, reagents, equipment, and instruments. The “*action*” category describes operations performed by or to the entity. An “*observation*” consists of empirical data captured during the course of an experiment; this can include both structured machine-generated data files as well as unstructured general comments from the human operator. An “*outcome*” is a subcategory of observation describing a (intermediate) product; flagging these as possible dependent variables facilitates use by data scientists. Additional discussion of categories can be found in the Supplementary Information (Fig. S1).

Capture

Subsequent sections of this article detail the underlying software process of ESCALATE for metal halide perovskite data capture and reporting. There are five key data capture steps positioned throughout the pipeline. Each section of data capture has been designed with extensibility, flexibility, and with a regime for reporting in mind. The points discussed in the following sections should provide sufficient background to extend ESCALATE to other materials investigations. ESCALATE is

designed as a lightweight way to collect comprehensive data from existing and new experiments, and simplify the data capture, reporting, and transition to “closed loop” operation. ESCALATE is equally suited to any combination of human- and automated processes, unlike comprehensive self-driving laboratory software designed for fully automated operation,^[41,62] and well-tailored high-throughput infrastructures for specific materials problems.^[27] Rather than providing a single implementation strategy, ESCALATE facilitates data reporting and experiment specification through a human- and machine-interpretable file infrastructure compatible with most machine learning programming languages. The key forms, as well as a brief description of the operation of the pipeline software, are outlined in Fig. 2.

Specify

The first point of data collection involves capturing user-provided information to initialize the software (Fig. 2). These data are descriptions of the *type* of experiment, including what is generally referred to as the metadata of the entity. ESCALATE uses an executable script to capture initial user specifications. ESCALATE parses the provided constraints which include chemicals, reagent, and custom constraints on the experimental entity. An example of the data captured in the specification step of ESCALATE is provided in Table II.

Table II. Examples of the structure of data collection necessary to successfully generate perovskite experimental template file.

Category	Content	Entry category	Entry example
Type	Run overview	Run ID, experimenter name, number of experiments, notes	2018-07-05T15_03_19.157863+00_00_LBL (yyyy-mm-ddTHH_MM_SS.SSSSSS +UTCLabel_Lab)
Type	Version control	Software version, experimental protocol name, process ID	Version 1.0, Perovskite 1.0
Materials	Equipment	Reaction Vessels, Instrumentation	Vials, plates, equipment, instruments UV-vis, IR, XRD
Materials	Chemicals	Chemical Identifier	InChI, InChIKey, Canonical SMILES
Materials and actions	Reagents	Reagent identifier, reagent formulation with procedure	{Reagent ID: ##, chemicals: [{InChI: ##, nominal_amount: ##, actual_amount: ##}, { ... }], instructions: [{op: stir, duration: 0.5, duration_units: hr}, { ...}], operator: name, { ...}}

The experimental template constrains the experimental system and imposes requirements on data needed to define the subsequent models, specifically: a unique identifier (UID), version control for software and experimental process chemistry, and placeholders for the input reagents and chemical identifiers, and laboratory equipment/instrumentation needed. Requiring these data as part of specifying a template ensures that a common baseline of data exists for similar templates.

Chemicals and reagents have locally-resolved UIDs. For example, a URL accessible tabulated data entry form (such as those provided by an online spreadsheet) can store the list of the chemical components of a given reagent model along with an associated UID. In our ontology, “chemicals” describe pure compounds. We use InChIKeys; other identifiers (e.g., SMILES, InChI strings, CAS numbers, PubChem ID) can be used by ESCALATE, but in practice these have problems with canonical order and copyright. In principle, chemical provenance could be further tracked to specific laboratory inventory items, as is done in Aquarium,^[49] but this adds too much complexity to the early stages of data capture. “Reagents” describe a set of chemicals along with preparation instructions expressed as an extension to Autoprotocol^[46] (see Table II). Each reagent is assigned a UID, which allows for comparison across experiments. Competing approaches for assigning UIDs for reagents to UIDs are also available, such as Synbiohub for biologic components,^[63] or proposed IUPAC systems, such as RInChI or MInChI for descriptions of reactions and mixtures,^[64] respectively, but these only track composition and do not specify preparation conditions such as temperature and mixing parameters.

Equipment and instrumentation provide the last pieces of information needed to constrain the generation of models from a template. In the metal halide perovskite chemistry, the selection of the equipment determines the number of parallel reactions (i.e., how many wells), as well as setting baselines for nominal reaction conditions in the model. For instance, a different type of reaction plate, one being an aluminum block

and the other being steel, would exhibit different thermal properties across the plate. This, in turn, would affect the nominal temperature of each reaction well, and thereby alter the nominal temperature proposed for a given experiment at a particular location on the plate. Assigning an equipment UID provides another layer of necessary information for accurately capturing the intention of the experiment and generating accurate nominal models.

Depending on the experiment, other metadata could be tied to the template selection and initial model generation. For instance, the location where the experiment is performed, the chemist executing the code, and a notes section are reasonable data to capture when implementing a template for the generation of models.

Generate

Valid experiments must obey experiment-specific technical constraints (e.g., the total volume of reagents added cannot exceed the container volume) and composition-dependent physical constraints (e.g., the final concentration of a species cannot be greater than the most concentrated reagent solution). ESCALATE enforces these limits using the experiment template and reagent model description. For a given template, the chemical space is bounded hierarchically by the following constraints: (1) the *material* limits (i.e., the size and mass constraints of the equipment); (2) the *type* of reagent, and thereby the concentration of chemicals in the reagents; and (3) user-defined constraints (i.e., user-defined minimum concentration of a given chemical). Each additional constraint shrinks the potential search space, which can be quantified through the volume of the convex hull encapsulating possible points in chemical space. For instance, the lower concentration boundary for the organic component of a metal halide perovskite experiment is defined not by the binary mixture containing the organic and solvent, but by the ternary stock solution of inorganic (PbI₂), organic (e.g., ethyl ammonium iodide needed to solubilize the inorganic), and solvent (e.g., γ

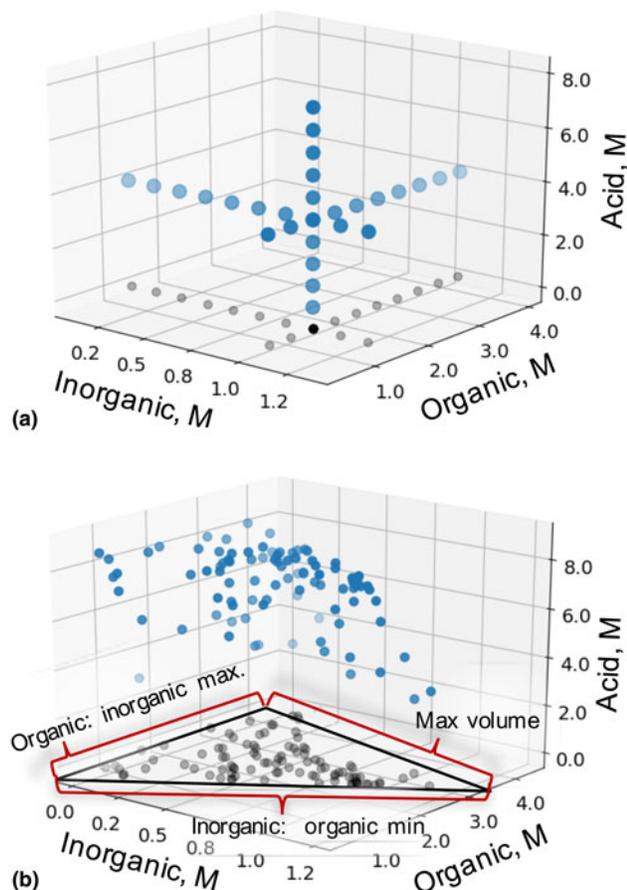


Figure 3. (a) Single Variable Modulation and (b) quasi-random sampling of metal halide perovskite chemical space in three dimensions. Constraints are illustrated for quasi-random sampling.

butyrolactone). To satisfy the dependency of the organic concentration originating from two reagents, ESCALATE samples the reagent volumes starting from the reagent with the most chemical components with conflicts or exceptions resolved by user specification. In this case, the volume of the ternary inorganic mixture is sampled first, followed by the binary organic mixture containing solvent and the organic, with the solvent volume sampled last. ESCALATE incorporates limits from each of the entities provided by the template, analyzes the boundaries through the described hierarchy [Fig. 3(b)], and samples from the maximum accessible convex hull to generate the desired number of experimental models.

ESCALATE provides two modes of experiment generation: (1) quasi-random sampling of a limited number of experiments within the user-specified constraints [Fig. 3(b)] and (2) exhaustive generation of the possible experimental “state space” suitable for use by external experiment recommendation programs (Fig. 4).

Quasi-random experiments generation

Efficient exploration of the multidimensional composition space is a challenging problem. The naive human strategy is

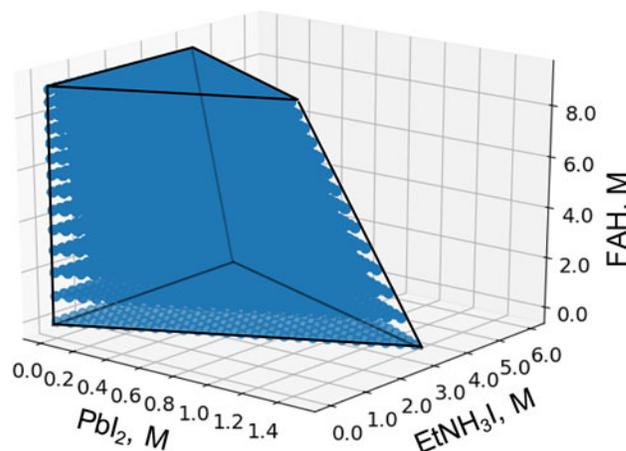


Figure 4. Visual representation of a reduced density state space for PbI_2 , EtNH_3I , and formic acid metal halide perovskite system.

to alter a single variable at a time [Fig. 3(a)], but this is inefficient and does not capture multivariate changes.^[24,31] Full or partial factorial grid searches are commonly employed in the statistical design of the experiments, but have the disadvantage of requiring prior knowledge of grid spacing and variable interaction orders. During the first exploration of a new chemical space, an expedient strategy is to generate a quasi-random distribution of compositions [Fig. 3(b)], which unlike pseudo-random sequences, minimize the discrepancy between sub-intervals.^[65] No prior knowledge of the relevant spacings is needed and each well composition is generated independently. This has the twofold advantage of (i) avoiding systematic spurious correlations between the position of the well (e.g., inconsistent heating or shaking) and the well composition and (ii) allowing an arbitrary number of experiments to be generated that adequately sample the accessible composition space.

Exhaustive experimental state space generation

A fully sampled convex hull defines the complete experimental “state space.” A file of this “state-set” provides non-domain expert collaborators with the full set of feasible experiments. ESCALATE generates a state space file containing the full reaction description (e.g., concentrations, physicochemical properties, etc.) to the operational specifications needed to define the experiment model (e.g., masses and volumes of chemicals needed). This enables machine learning algorithms to work with the full description of the experiment as an input, without having to carry out these calculations separately. An example of the state space for the PbI_2 , EtNH_3I , and formic acid metal halide perovskite system is shown in Fig. 4.

The state-set CSV file for PbI_2 , EtNH_3I , and formic acid system (three stock solution volume choices) is generated such that each point in the “state-set” refers to unique experiment. The estimated liquid handler dispensing precision is $\pm 5 \mu\text{L}$ resulting in $10 \mu\text{L}$ spacing between points along a

given reagent axis in the state space.^[66] In addition to the three reagent concentrations, the values of 67 reaction descriptors are included for each experiment in the “*state-set*.” The “*state-set*” generation requires approximately 15 s on a modern desktop computer and generates an approximately 200 megabyte file. At larger numbers of variables, the increase in computational resources required to generate and store the “*state-set*” becomes prohibitive; a solution to this scaling problem is on-demand validation of the proposed experiment plans, and is currently under development.

The *state-set*, along with curated and normalized data from previous experiments (i.e., training set), provides a means for outside collaborators, using their own preferred machine-learning strategies, to recommend possible reactions to the experimental process. Using a *state-set* as an intermediary enables participation without requiring a collaborator to understand specific experiment constraints or develop a mapping from reagent volume to chemical concentration. A non-expert user or artificial intelligence program only needs to recommend an experiment model from the state space to ensure that the constraints of the experimental template are met and that the object will be incorporated into the pipeline.

Prepare

After performing the experiment generation, described above, ESCALATE has defined a nominal model for each of the entities (reagent, chemicals, and experiments as in Fig. 1) which can be fully characterized by the *type*, *material*, and *actions* categories. The generated experimental models include all information necessary to fully describe the nominal experiments, including reaction temperatures, mixing times, equipment descriptions, specific locations on the robotic plate, and reagent dispensing data.

ESCALATE next sums the target dispense volume of each reagent for all generated experiments occurring on a single 96-well plate and calculates the total nominal amount of each chemical needed to prepare a sufficient quantity of each reagent. Chemical information required to calculate unit conversions for presentation to the operator (e.g., molecular weight, density) is centrally stored in a human interfaceable web-hosted spreadsheet.

The reagent models along with the nominal preparation instructions are provided to the operator as an interactive form with specific handling instructions (temperature, stir rate, duration, etc.) along with nominal amounts of each chemical necessary to prepare the reagents. More generally, interactive forms provide a means to expose intermediary data during ESCALATE operation. The reagent model/object interface provides a single repository for both the reagent data calculated during experimental model generation as well as observations about the reagent objects prepared in the laboratory. In particular, the shared form minimizes human error during entry by automatically prefilling available sections and highlighting to the operator the sections for recording materials, observations, and notes.

A simplified example of the interactive data-entry form is shown in Fig. 5. A full example of the metal halide perovskite interactive form file is discussed in the Supplementary Information. A more detailed illustration is also included as Fig. S2. The completed form is converted to JSON format,^[67] a human readable and computer interpretable format, for later data workup and post-processing.

Execute

Execution of experiment objects generated by ESCALATE has been predominantly performed by a Hamilton Microlab NIMBUS4 robot. The current configuration of the Hamilton software uses an XLS formatted control file to instruct the volume dispensed into each of the 96 wells on a plate. For our particular purposes, ESCALATE has been programmed to output experimental models as a spreadsheet, but the final output of the experimental object file is flexible. A subset of the experimental model file is provided in Table III and a specific example file is discussed in the Supplementary Information (Experiment-Model_RobotInput.xls).

Note that the experimental model file includes *type* information, relevant for describing the particular location on the plate an experiment is being performed. Nominal reagent dispensing information is dictated to the robot on a well-by-well basis and can vary from one experiment to the next. Other properties such as reaction time, mixing conditions, and temperature are nominal values that apply to the reaction plate as a whole; the robot does not have the ability to mix individual wells at different rates.

ESCALATE has also been used in combination with human operators for reagent dispensing with little alteration of the code. For instance, manually dispensing highly varied volumes to specific locations on a plate is tedious. To ease implementation for human interaction, manual modification of the nominal reagent volumes after generation of the experimental model is fully supported. Thereby, an operator can alter the experimental model file if necessary to accurately capture the intention of the experiments.

Observe

Observations collected during the perovskite experiments include ambient temperature and humidity, thermal images which map the actual temperature across the heating block, operator notes, liquid handler operational log files, instrument configuration and setting files, images of the products, and the final experimental outcome. Instrument output files are copied directly to the online repository for later use and human observations are recorded by the operator during the experiment. Metadata capture and management solutions for instrumentation, such as Allotrope Standard,^[68] can be included as files in the ESCALATE data lake.

Other files collected during ESCALATE operation, including parsable outputs from instruments, images, and log files, are attributed to the relevant experiments. The advantage of a data lake architecture is that no initial effort needs to be

		Run Data	Reagent Preparation Information		
Date Created	2018-09-13	Reagent	Temp (C)	Stir (RPM)	
Time Created UTC	14_00_50	1	<i>null</i>	<i>null</i>	
Laboratory	LBL	2	<i>75</i>	<i>450</i>	
Operator Name	<i>Zhi Li</i>	3	<i>75</i>	<i>450</i>	
Exp Workflow Ver	1.10				
Generator Workflow Ver	1.20				
Notes	Experiments performed based challenge problem recommendations				
Experimental Summary:	No modifications to workflow 1 protocol				
	Chemical Abbreviation	Nominal (Ideal) Amount	Actual (Empirical) Amount	Measurement Unit	
Reagent 1	GBL	20.6	<i>30.0</i>	milliliter	
Reagent 2	Final Volume =	27.2	<i>37.5</i>	milliliter	
Chemical 1	Pbl2	18.8	<i>18.7</i>	gram	
Chemical 2	EtNH3I	14.1	<i>14.2</i>	gram	
Chemical 3	GBL	27.2	<i>27.5</i>	milliliter	
Reagent 3	Final Volume =	18.7	<i>17.0</i>	milliliter	
Chemical 1	EtNH3I	9.5	<i>9.6</i>	gram	
Chemical 2	GBL	9.2	<i>9.2</i>	milliliter	

Figure 5. Representation of the simplified reagent model/object interface for capturing metal halide perovskite reagent preparation. Highlighted sections are observations required for executing subsequent steps of the pipeline, all other cells are filled by ESCALATE.

invested in parsing them before they can be collected. However, by storing them all, should scientific interest arise, a future processing pipeline can retroactively extract the relevant data. A full example of the data collected during typical ESCALATE operation is included as and discussed in the Supplementary Information.

Report

The second portion of code processes the gathered data into a structured format that can be used for analysis (e.g., data visualization, machine learning), as depicted in Fig. 6. Categories of data (i.e., *material*, *action*, etc.) can be individually targeted by coding operations designed for converting grouped data into the final report. For example, the final concentration of a

particular chemical in an experiment is dependent upon empirical measurements contained in the reagent model and the experimental model, two large repositories of data. However, the *material* category contains all of the information necessary to characterize the concentration of a specific chemical in the final experiment. More importantly, the organization of the data into the *material* category provides a target staging platform for the development of automated data processing pipelines. The categories are thereby effective not only from the perspective of ensuring accurate data capture, but also for providing an extensible platform to apply ESCALATE to new classes of experiments.

The report format depends upon the recipient and the type of experiment. For shallow datasets with a limited total number of

Table III. Example of selected entries taken from the experimental models' file (robot input file) including the nominal values for entities (vial site, labware), materials (reagents), actions (temperature, mixing).

Vial Site	labware ID	Reagent 1 (μL)	Reagent 2 (μL)	Reagent 3 (μL)	Reaction temperature (C)	Reaction time (s)	Mix rate (rpm)	Mix time (s)	Reagent 1 temperature (C)
A1	Symyx_96_well	402	50	48	70	12,600	750	900	45
B1	Symyx_96_well	78	29	393	70	12,600	750	900	45
-	-	-	-	-	-	-	-	-	-
H12	Symyx_96_well	211	114	175	70	12,600	750	900	45

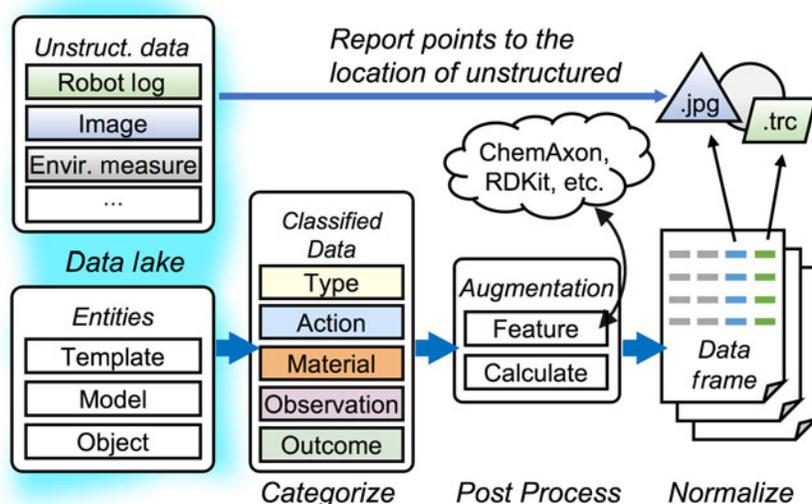


Figure 6. ESCALATE process for generating the final report data frame.

experiments, as is common in materials chemistry research, a two-dimensional data array file (e.g., CSV) is readily imported by data processing packages. Data presented in two dimensions are often wide (i.e., large numbers of columns) which encumber human inspection and analysis without the aid of additional software. Employing systematic naming schemes can aid in the processing of 2d data. The implementation used in ESCALATE is outlined in Table IV.

The key features of the naming syntax include (1) keyword-oriented titles; (2) informative, machine-parsable prefixes and suffixes; (3) avoiding spaces. Keywords provide a general description of what is contained in a particular column of data. If a user or program needs to operate on all “chemical” and “InChIKey” information, then the column headers must be similar enough to be generally referenced. Further specification by the user could include a numerical argument, “reagent_1_chemical_1_InChIKey” providing only the InChIKey from a specific chemical in “reagent 1.”

Data presentation generated by ESCALATE uses prefixes to help parse different general categories of data. “Raw” refers to unprocessed data that are often unsuitable for consumption by non-domain experts. An example is extensive quantities such as mass that specify a particular experiment, even though the intensive property of concentration is more relevant. “Rxn” includes both processed and unprocessed data experimental specifications that are suitable for machine learning and data analysis. “Feat” and “calc” describe the physicochemical properties of the chemicals and reagent mixtures, respectively, and are calculated using ChemAxon^[69] and RDKit^[70] in the current implementation. “Out” indicates suitable dependent variables to be predicted by machine learning applications (e.g., metal halide perovskite crystal score). Suffixes distinguish “nominal” (i.e., experimental intent) and “actual” (i.e., empirical observation) values. Using an underscore or hyphen delineated division between prefix, description, and suffix avoids common problems with parsing as some programming languages handle spaces poorly.

Table IV. Overview of the namespace for dataset normalization.

Header Prefix/suffix	Description of header	Example
raw*	Unprocessed raw data from pipeline	_raw_reagent_3_conc
rxn*	Processed data from pipeline	_rxn_M_organic_actual
feat*	Molecular features (various sources*)	_feat_VDWSurfaceArea
out*	Processed <i>output</i> data from pipeline (model targets)	_out_crystalscore
calc*	Calculated features based on processed experimental data	_calc_avgNVDWVol
*_actual_	Measured chemical, reagent, and experimental properties	_rxn_M_inorganic_actual
*_nominal_	Proposed chemical, reagent, and experimental properties	_rxn_M_acid_nominal

* is a wildcard indicating any combination of characters, phrases, or abbreviations are acceptable at the indicated position.

Table V. Final data-frame structured for distribution.

RunID_vial	_out _crystalscore	_rxn_M_organic _actual	_rxn_C_temperature _actual	_feat_*	_calc_*	_raw_*
2017-10-16T17_52_59.000000 +00_00_LBL_A1	1	2.924	77	##	##	##
–	–	–	–	–	–	–
2018-12-06T17_08_57.099365 +00_00_LBL_H3	2	0.826	67	##	##	##
2018-12-06T17_08_57.099365 +00_00_LBL_H4	4	1.571	67	##	##	##
2018-12-06T17_08_57.099365 +00_00_LBL_H5	1	4.180	67	##	##	##

* is a wildcard indicating any combination of characters, phrases, or abbreviations are acceptable at the indicated position.

A simplified example of an ESCALATE rendered report in the format of a 2d data frame is presented in Table V, a full example from a single plate of reactions is included as FinalReport_2d_Curated.csv with the Supplementary Information. Each row of the final data frame represents a complete description of the experimental intent and the performance of the nominal experiment. The namespace, “RunID_Vial,” is a concatenation of the well site in which the reaction was performed in combination with the previously described “Run ID.” The namespace represents the UID of the experiment; no duplicates should exist in the final dataset. The headers contain the relevant prefixes and suffixes to describe the data and are organized in such a way to group similar data.

ESCALATE curates experimental data to support “best-practices” such as the F.A.I.R. data principles,^[71] which calls for making the data Findable, Accessible, Interoperable, and Reusable. ESCALATE makes the data *findable* by assigning each experiment a unique human- and machine-readable identifier to maintain origin, laboratory, operator, and date provenance to each entry. ESCALATE is not a publication/dissemination platform, but broader *accessibility* can be achieved by upload to public repositories such as NREL’s HTEM DB^[37] or Citrine.^[72] *Interoperability* is provided by the report generation which compiles the experimental data and metadata into a CSV format, which can be imported into a wide variety of visualization, statistics, and machine learning packages. The report also incorporates calculations of physicochemical properties of the reagents, stoichiometry, concentration, etc., that facilitate the use of these data by non-domain experts. The data are *reusable* at two levels. The output reports described above are reusable for various analyses. Furthermore, the metadata persistently associated with each item in the data lake allows for retroactive retrieval and analysis of other experimental details should a future need arise.

Closed-loop operation

The current infrastructure for closed-loop operation uses two CSV files in a shared directory to provide the *report* of past

experimental results that can be used to train the model and the *state-set* of possible experiments. The *state-set* includes a complete description of each nominal model experiment, including all of the computed material physicochemical features and stoichiometric calculations that are included in the report. By providing all of the necessary domain-specific calculations, the report and state-set files allow non-domain experts to easily import the existing metal halide perovskite dataset, train a model using existing data, and make recommendations for upcoming experiments. Experiment recommendations are communicated by writing a file containing a list of desired state-set indices into a shared directory. An ESCALATE command line program reads these files and generates the complete experiment description (experiment template, experiment model, reagent model, etc.) to be performed by the operator. ChemOS uses a similar shared-filed based scheme for closed-loop operation.^[41] An experiment submission and validation API is currently under development.

Results

ESCALATE has been operating for 10 months as of April 2019. In that time, the data pipeline has operated a total of 55 times, capturing a total of 4336 single-crystal metal halide perovskite reactions. The throughput volume of data capture using this software is illustrated in Fig. 7(a).

The throughput of data collection has only been limited by experimental capacity; ESCALATE has facilitated the maximum bandwidth of experiments possible. These data represent one of the largest metal halide perovskite crystallization datasets ever curated. The comprehensive data capture pipeline results in a dataset containing all of the “dark reactions”—otherwise unreported “failures” and marginal success—that are crucial for the development of machine learning models in materials science research.^[20] The template-driven experiment generation results in an unbiased sampling of chemical space. The ontology and data structure developed for ESCALATE has also enabled collaboration with non-domain experts. Four

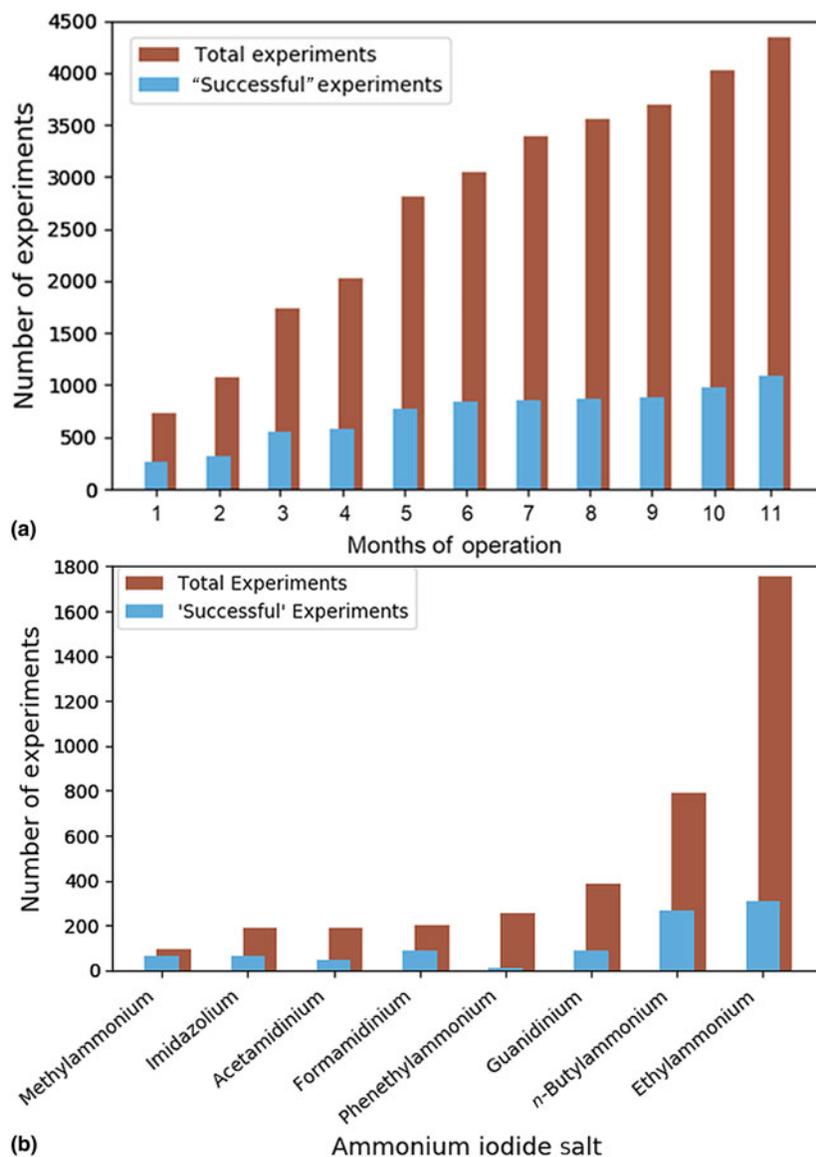


Figure 7. (a) Total experimental throughput of curated experiments captured by ESCALATE present in the final dataset by month; (b) number of experiments performed for different organic cations (ammonium iodide salt).

research groups—at Lawrence Berkeley National Laboratory, the Broad Institute, Haverford College, and Netrias—comprised of a mixture of chemists, materials scientists, computer scientists, and mathematicians—have recommended more than 96 metal halide perovskite experiments per week generated by a variety of machine learning algorithms trained on ESCALATE generated *report* CSV files. These recommended experiments are conducted at the Lawrence Berkeley National Laboratory’s Molecular Foundry, resulting in a total of 16 iterations of a DARPA sponsored interactive campaign for materials discovery. This campaign is ongoing, and results will be reported in future articles.

Various experimental workflows have been developed in the laboratory and implemented using ESCALATE for data

capture. The most successful workflow has covered eight different single-crystal metal halide perovskite experiments using lead iodide, an ammonium iodide, and formic acid in GBL [Fig. 7(b)]. Work is underway to further broaden ESCALATE to aid in the exploration of metal halide perovskite chemical space. Specifically, we aim to incorporate additional halides, amines, and inorganics and various crystallization regimes into this dataset in the near future.

Current limitations

Limitations of the current version of ESCALATE reflect the experimental needs of the perovskite experiments discussed above, in which up to seven human-prepared solutions of reagents (comprised of up to three chemicals) are dispensed

as solutions either by a human operator or robot in batches of no more than 96 experiments. (The underlying data handling supports an unlimited number of reagents and chemicals, but the current ESCALATE generated graphical user interface is limited to up to three chemicals per reagent for the human operator instruction, and up to seven reagents per experiment for the liquid handler control file. The other core functionalities of ESCALATE—specification, generation, preparation, observation, and reporting—support input of any number of chemicals per reagent composition, number of reagent solutions, and number of experiments.) In the example described in this article, ESCALATE generates the experimental model file as an XLS in the format used by a Hamilton Microlab NIMBUS4 robot. The experimental model file used to operate the NIMBUS liquid handler robot also includes the robotic execution control file specific to this instrument and experiment type. These instrument- and vendor-specific programs can be modified by the users for the desired application and instrumentation available; a copy is retained in the data lake for each experiment.

The data handling is performed using pandas dataframes which have a variety of native output formats^[73] that allow the experimental model, reagent model, and final report files to be rendered as a CSV, textfile, XLSX, or other standard formats via minor modifications of the code; exceptions to the pandas data handling occurs only in intermediary file storage mechanisms (e.g., Autoprotocol formatted JSON). Closed-loop execution is initiated by executing a command line program that reads the recommended experiments from the *state-set* CSV, similar to the operation of ChemOS.^[41]

Concluding remarks

ESCALATE is both an ontologic framework for describing experimental entities in a computer-friendly format and a software package for facilitating experiment specification, data capture, and reporting. The initial development of infrastructure can be a daunting task for scientists unfamiliar with the process. Often the simplest course of action is to start by organizing the relevant *types*, *materials*, *actions*, *observations*, and *outcomes* needed to describe the target system. This article describes examples of data belonging to these categories and demonstrates how to assemble meaningful datasets from typical laboratory processes, organized around familiar chemical concepts and structured to facilitate both human and machine interpretability. We offer a specific demonstration to single-crystal metal halide perovskite synthesis, which illustrates laboratory manipulations, high-throughput applications, and interfacing challenges common to many research endeavors. The combination of this ontology with the software package makes it easy for experimentalists to start collecting valuable, comprehensive datasets in automated and semi-automated laboratory environments without a large initial investment in database design. Additionally, a non-expert user or artificial intelligence program need only recommend an experiment generated by ESCALATE to ensure that the constraints of the chemical

system are met and can be performed in the laboratory. Captured files are stored in a way that facilitates transparent access by users, even those without programming background. At the same time, the structure imposed by the ontology facilitates future data extraction. Future work will be aimed at improving data workflow, generalizing ESCALATE to additional experimental systems, and closed-loop autonomous experimental design.

Code availability

The code used for this project can be found at the following links: https://github.com/darkreactions/ESCALATE_Capture and https://github.com/darkreactions/ESCALATE_report and is released under an MIT license. Discussion about code implementation and a comprehensive overview of a single iteration of ESCALATE along with the associated files can be found in the Supplementary Information.

Supplementary material

The supplementary material for this article can be found at <https://doi.org/10.1557/mrc.2019.72>.

Acknowledgments

We thank Alex Cristofaro (MIT Broad Institute) and Scott Novotney (Two-Six Labs) for helpful feedback in the development of the ESCALATE state space and file reporting mechanisms. This material is based upon the work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001118C0036. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. This material is DISTRIBUTION A. Approved for public release: distribution unlimited. Work at the Molecular Foundry was supported by the Office of Science, Office of Basic Energy Sciences, of the US Department of Energy under Contract No. DE-AC02-05CH11231. J.S. acknowledges the Henry Dreyfus Teacher-Scholar Award (TH-14-010).

References

1. NSF CHE Workshop: Framing the Role of Big Data and Modern Data Science in Chemistry. Available at: https://www.nsf.gov/mps/che/workshops/data_chemistry_workshop_report_03262018.pdf (accessed December 21, 2018).
2. Mission Innovation: Materials Acceleration Platform: Accelerating Advanced Energy Materials Discovery by Integrating High-Throughput Methods with Artificial Intelligence Report of the Clean Energy Materials Innovation Challenge Expert Workshop. Available at: <http://mission-innovation.net/wp-content/uploads/2018/01/Mission-Innovation-IC6-Report-Materials-Acceleration-Platform-Jan-2018.pdf> (accessed December 21, 2018).
3. Multi-Agency, Multi-Year Program Plan in Advanced Energy Materials Discovery, Development, and Process Design: Available at: https://www.energy.gov/sites/prod/files/2018/12/f58/Multi-Agency%20Multi-Year%20Program%20Plan%20in%20Advanced%20Energy%20Materials%20Discovery%20Development%20and%20Process%20Design_Workshop%20Summary%20Report.pdf (accessed December 21, 2018).
4. A.B. Henson, P.S. Gromski, and L. Cronin: Designing algorithms to aid discovery by chemical robots. *ACS Cent. Sci.* **4**, 793–804 (2018).

5. D.P. Tabor, L.M. Roch, S.K. Saikin, C. Kreisbeck, D. Sheberla, J.H. Montoya, S. Dwaraknath, M. Aykol, C. Ortiz, H. Tribukait, C. Amador-Bedolla, C.J. Brabec, B. Maruyama, K.A. Persson, and A. Aspuru-Guzik: Accelerating the discovery of materials for clean energy in the era of smart automation. *Nat. Rev. Mater.* **3**, 5–20 (2018).
6. J.-P. Correa-Baena, K. Hippalgaonkar, J. van Duren, S. Jaffer, V.R. Chandrasekhar, V. Stevanovic, C. Wadia, S. Guha, and T. Buonassisi: Accelerating materials development via automation, machine learning, and high-performance computing. *Joule* **2**, 1410–1420 (2018).
7. X.-D. Xiang, X. Sun, G. Briceño, Y. Lou, K.-A. Wang, H. Chang, W.G. Wallace-Freedman, S.-W. Chen, and P.G. Schultz: A combinatorial approach to materials discovery. *Science* **268**, 1738–1740 (1995).
8. P.G. Schultz and X.-D. Xiang: Combinatorial approaches to materials science. *Curr. Opin. Solid State Mater. Sci.* **3**, 153–158 (1998).
9. H. Koinuma and I. Takeuchi: Combinatorial solid-state chemistry of inorganic materials. *Nat. Mater.* **3**, 429–438 (2004).
10. I. Takeuchi, R.B. van Dover, and H. Koinuma: Combinatorial synthesis and evaluation of functional inorganic materials using thin-film techniques. *MRS Bull.* **27**, 301–308 (2002).
11. Z.H. Barber and M.G. Blamire: High throughput thin film materials science. *Mater. Sci. Technol.* **24**, 757–770 (2008).
12. S.I. Woo, K.W. Kim, H.Y. Cho, K.S. Oh, M.K. Jeon, N.H. Tarte, T.S. Kim, and A. Mahmood: Current status of combinatorial and high-throughput methods for discovering new materials and catalysts. *QSAR Comb. Sci.* **24**, 138–154 (2005).
13. M.L. Green, I. Takeuchi, and J.R. Hattrick-Simpers: Applications of high throughput (combinatorial) methodologies to electronic, magnetic, optical, and energy-related materials. *J. Appl. Phys.* **113**, 231101 (2013).
14. L.A. Baumes, P. Serna, and A. Corma: Merging traditional and high-throughput approaches results in efficient design, synthesis and screening of catalysts for an industrial process. *Appl. Catal. A* **381**, 197–208 (2010).
15. R. Potyrailo, K. Rajan, K. Stoeve, I. Takeuchi, B. Chisholm, and H. Lam: Combinatorial and high-throughput screening of materials libraries: review of state of the art. *ACS Comb. Sci.* **13**, 579–633 (2011).
16. M. Shevlin: Practical high-throughput experimentation for chemists. *ACS Med. Chem. Lett.* **8**, 601–607 (2017).
17. W.F. Maier, K. Stöwe, and S. Sieg: Combinatorial and high-throughput materials science. *Angew. Chem. Int. Ed Engl.* **46**, 6016–6067 (2007).
18. K.T. Butler, D.W. Davies, H. Cartwright, O. Isayev, and A. Walsh: Machine learning for molecular and materials science. *Nature* **559**, 547–555 (2018).
19. B. Sanchez-Lengeling and A. Aspuru-Guzik: Inverse molecular design using machine learning: generative models for matter engineering. *Science* **361**, 360–365 (2018).
20. P. Raccuglia, K.C. Elbert, P.D.F. Adler, C. Falk, M.B. Wenny, A. Mollo, M. Zeller, S.A. Friedler, J. Schrier, and A.J. Norquist: Machine-learning-assisted materials discovery using failed experiments. *Nature* **533**, 73–76 (2016).
21. D.T. Ahneman, J.G. Estrada, S. Lin, S.D. Dreher, and A.G. Doyle: Predicting reaction performance in C-N cross-coupling using machine learning. *Science* **360**, 186–190 (2018).
22. S. Lin, S. Dikler, W.D. Blincoe, R.D. Ferguson, R.P. Sheridan, Z. Peng, D.V. Conway, K. Zawatzky, H. Wang, T. Cernak, I.W. Davies, D.A. DiRocco, H. Sheng, C.J. Welch, and S.D. Dreher: Mapping the dark space of chemical reactions with extended nanomole synthesis and MALDI-TOF MS. *Science* **361**, eaar6236 (2018).
23. R.J. Xu, J.H. Olshansky, P.D.F. Adler, Y. Huang, M.D. Smith, M. Zeller, J. Schrier, and A.J. Norquist: Understanding structural adaptability: a reactant informatics approach to experiment design. *Mol. Syst. Des. Eng.* **3**, 473–484 (2018).
24. V. Duros, J. Grizou, W. Xuan, Z. Hosni, D.-L. Long, H.N. Miras, and L. Cronin: Human versus robots in the discovery and crystallization of gigantic polyoxometalates. *Angew. Chem. Int. Ed. Engl.* **56**, 10815–10820 (2017).
25. Z. Zhou, X. Li, and R.N. Zare: Optimizing chemical reactions with deep reinforcement learning. *ACS Cent. Sci.* **3**, 1337–1344 (2017).
26. A.-C. Bédard, A. Adamo, K.C. Aroh, M.G. Russell, A.A. Bedermann, J. Torosian, B. Yue, K.F. Jensen, and T.F. Jamison: Reconfigurable system for automated optimization of diverse chemical reactions. *Science* **361**, 1220–1225 (2018).
27. P. Nikolaev, D. Hooper, F. Webber, R. Rao, K. Decker, M. Krein, J. Poleski, R. Barto, and B. Maruyama: Autonomy in materials research: a case study in carbon nanotube growth. *npj Comput. Mater.* **2**, 16031 (2016).
28. A.G. Kusne, T. Gao, A. Mehta, L. Ke, M.C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M.J. Kramer, C. Long, and I. Takeuchi: On-the-fly machine-learning for high-throughput experiments: search for rare-earth-free permanent magnets. *Sci. Rep.* **4**, 6367 (2014).
29. B. Celse, S. Rebours, F. Gay, P. Coste, L. Bourgeois, O. Zammit, and V. Lebacque: Integration of an informatics system in a high throughput experimentation. Description of a global framework illustrated through several examples. *Oil Gas Sci. Technol.—Rev. IFP Energies nouvelles* **68**, 445–468 (2013).
30. J. Bai, Y. Xue, J. Bjorck, R. Le Bras, B. Rappazzo, R. Bernstein, S.K. Suram, R.B. Van Dover, J.M. Gregoire, and C.P. Gomes: Phase mapper: accelerating materials discovery with AI. *AI Mag* **39**, 15 (2018).
31. B. Cao, L.A. Adutwum, A.O. Oliynyk, E.J. Lubner, B.C. Olsen, A. Mar, and J. M. Buriak: How To optimize materials and devices via design of experiments and machine learning: demonstration using organic photovoltaics. *ACS Nano* **12**, 7434–7444 (2018).
32. V. Stanev, C. Oses, A.G. Kusne, E. Rodriguez, J. Paglione, S. Curtarolo, and I. Takeuchi: Machine learning modeling of superconducting critical temperature. *npj Comput. Mater.* **4**, 1 (2018).
33. Q. Yan, J. Yu, S.K. Suram, L. Zhou, A. Shinde, P.F. Newhouse, W. Chen, G. Li, K.A. Persson, J.M. Gregoire, and J.B. Neaton: Solar fuels photoanode materials discovery by integrating high-throughput theory and experiment. *Proc. Natl. Acad. Sci. USA* **114**, 3040–3043 (2017).
34. F. Ren, L. Ward, T. Williams, K.J. Laws, C. Wolverton, J. Hattrick-Simpers, and A. Mehta: Accelerated discovery of metallic glasses through iteration of machine learning and high-throughput experiments. *Sci. Adv.* **4**, eaaq1566 (2018).
35. A. Shinde, S.K. Suram, Q. Yan, L. Zhou, A.K. Singh, J. Yu, K.A. Persson, J.B. Neaton, and J.M. Gregoire: Discovery of manganese-based solar fuel photoanodes via integration of electronic structure calculations, Pourbaix stability modeling, and high-throughput experiments. *ACS Energy Lett.* **2**, 2307–2312 (2017).
36. M.L. Green, C.L. Choi, J.R. Hattrick-Simpers, A.M. Joshi, I. Takeuchi, S.C. Barron, E. Campo, T. Chiang, S. Empedocles, J.M. Gregoire, A.G. Kusne, J. Martin, A. Mehta, K. Persson, Z. Trautt, J. Van Duren, and A. Zakutayev: Fulfilling the promise of the materials genome initiative with high-throughput experimental methodologies. *Appl. Phys. Rev.* **4**, 011105 (2017).
37. A. Zakutayev, N. Wunder, M. Schwarting, J.D. Perkins, R. White, K. Munch, W. Tumas, and C. Phillips: An open experimental database for exploring inorganic materials. *Sci. Data* **5**, 180053 (2018).
38. J. Li, Y. Lu, Y. Xu, C. Liu, Y. Tu, S. Ye, H. Liu, Y. Xie, H. Qian, and X. Zhu: AIR-Chem: authentic intelligent robotics for chemistry. *J. Phys. Chem. A* **122**, 9142–9148 (2018).
39. N. Adams and U.S. Schubert: From data to knowledge: chemical data management, data mining, and modeling in polymer science. *J. Comb. Chem.* **6**, 12–23 (2004).
40. N. Adams and U.S. Schubert: Software solutions for combinatorial and high-throughput materials and polymer research. *Macromol. Rapid Commun.* **25**, 48–58 (2004).
41. L.M. Roch, F. Häse, C. Kreisbeck, T. Tamayo-Mendoza, L.P.E. Yunker, J.E. Hein, and A. Aspuru-Guzik: ChemOS: orchestrating autonomous experimentation. *Sci Robot.* **3**, eaat5559 (2018).
42. J. Hachmann, M.A.F. Afzal, M. Haghighatdari, and Y. Pal: Building and deploying a cyberinfrastructure for the data-driven design of chemical systems and the exploration of chemical space. *Mol. Simul.* **44**, 921–929 (2018).
43. L.A. Baumes, S. Jimenez, and A. Corma: hiTeQ: a new workflow-based computing environment for streamlining discovery. Application in materials science. *Catal. Today* **159**, 126–137 (2011).
44. K. Tran, A. Palizhati, S. Back, and Z.W. Ulissi: Dynamic workflows for routine materials discovery in surface science. *J. Chem. Inf. Model.* **58**, 2392–2400 (2018).

45. M. Bates, A.J. Berliner, J. Lachoff, P.R. Jaschke, and E.S. Groban: Wet Lab accelerator: a web-based application democratizing laboratory automation for synthetic biology. *ACS Synth. Biol.* **6**, 167–171 (2017).
46. Autoprotocol: Available at: <http://autoprotocol.org/> (accessed January 8, 2019).
47. G. Linshiz, N. Stawski, S. Poust, C. Bi, J.D. Keasling, and N.J. Hillson: PaR-PaR laboratory automation platform. *ACS Synth. Biol.* **2**, 216–222 (2013).
48. E. Whitehead, F. Rudolf, H.-M. Kaltenbach, and J. Stelling: Automated planning enables complex protocols on liquid-handling robots. *ACS Synth. Biol.* **7**, 922–932 (2018).
49. B. Keller, J. Vrana, A. Miller, G. Newman, and E. Klavins: *Aquarium: The Laboratory Operating System (Version v2.5.0)*. Zenodo. (2019).
50. Emerald Cloud Lab: Available at: <https://www.emeraldcloudlab.com/> (accessed January 11, 2019).
51. B. Miles and P.L. Lee: Achieving reproducibility and closed-loop automation in biological experimentation with an IoT-enabled lab of the future. *SLAS Technol.* **23**, 432–439 (2018).
52. Transcriptic: Powering On-Demand Biology | Transcriptic. Available at: <https://transcriptic.com/> (accessed January 15, 2019).
53. D.B. Mitzi: Synthesis, Structure, and Properties of Organic-Inorganic Perovskites and Related Materials In *Progress in Inorganic Chemistry*, edited by K.D. Karlin (John Wiley & Sons, Inc., 9, Hoboken, NJ, USA, 1999), pp. 1–121.
54. M.D. Smith, E.J. Crace, A. Jaffe, and H.I. Karunadasa: The diversity of layered halide perovskites. *Annu. Rev. Mater. Res.* **48**, 111–136 (2018).
55. S. Li, C. Zhang, J.-J. Song, X. Xie, J.-Q. Meng, and S. Xu: Metal halide perovskite single crystals: from growth process to application. *Crystals (Basel)* **8**, 220 (2018).
56. H.J. Snaith: Present status and future prospects of perovskite photovoltaics. *Nat. Mater.* **17**, 372–376 (2018).
57. M.I.H. Ansari, A. Qurashi, and M.K. Nazeeruddin: Frontiers, opportunities, and challenges in perovskite solar cells: a critical review. *J. Photochem. Photobiol. C: Photochem. Rev.* **35**, 1–24 (2018).
58. F. Yao, P. Gui, Q. Zhang, and Q. Lin: Molecular engineering of perovskite photodetectors: recent advances in materials and devices. *Mol. Syst. Des. Eng.* **3**, 702–716 (2018).
59. G. Lozano: The role of metal halide perovskites in next-generation lighting devices. *J. Phys. Chem. Lett.* **9**, 3987–3997 (2018).
60. M.D. Smith and H.I. Karunadasa: White-light emission from layered halide perovskites. *Acc. Chem. Res.* **51**, 619–627 (2018).
61. S. Ahmad, C. George, D.J. Beesley, J.J. Baumberg, and M. De Volder: Photo-rechargeable organo-halide perovskite batteries. *Nano Lett.* **18**, 1856–1862 (2018).
62. F. Häse, L.M. Roch, and A. Aspuru-Guzik: Next-generation experimentation with self-driving laboratories. *TRECHEM*. Doi:10.1016/j.trechm.2019.02.007.
63. J.A. McLaughlin, C.J. Myers, Z. Zundel, G. Misirli, M. Zhang, I.D. Ofiteru, A. Goñi-Moreno, and A. Wipat: Synbiohub: a standards-enabled design repository for synthetic biology. *ACS Synth. Biol.* **7**, 682–688 (2018).
64. G. Grethe, G. Blanke, H. Kraut, and J.M. Goodman: International chemical identifier for reactions (RinChI). *J. Cheminform.* **10**, 22 (2018).
65. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling: *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. (Cambridge University Press, Cambridge, New York, 1992).
66. The precision of the NIMBUS4 is negatively impacted by the operating conditions required for metal halide perovskite synthesis including high temperature and use of GBL as a solvent.
67. JSON: Available at: <http://json.org/> (accessed January 11, 2019).
68. Allotrope Foundation Data Standard: Available at: <https://www.allotrope.org> (accessed January 15, 2019).
69. ChemAxon—Software Solutions and Services for Chemistry & Biology: Available at: <https://chemaxon.com/> (accessed 4 January 2019).
70. G. Landrum: RDKit, Available at: <http://www.rdkit.org> (accessed 15 January 2019).
71. M.D. Wilkinson, M. Dumontier, I.J.J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L.B. da Silva Santos, P.E. Bourne, J. Bouwman, A.J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C.T. Evelo, R. Finkers, A. Gonzalez-Beltran, A.J.G. Gray, P. Groth, C. Goble, J.S. Grethe, J. Heringa, P.A.C. 't Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S.J. Lusher, M.E. Martone, A. Mons, A.L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M.A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, and B. Mons: The FAIR guiding principles for scientific data management and stewardship. *Sci. Data* **3**, 160018 (2016).
72. Citrine Informatics: Available at: <https://citrine.io/> (accessed March 22, 2019).
73. W. McKinney: Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*, edited by S. van der Walt and J. Millman, (Scipy 2010, Austin, TX, 2010), pp. 51–56.