CRIPT: A Scalable Polymer Material Data Structure

Dylan J. Walsh[a], Weizhong Zou[a], Ludwig Schneider[b], Reid Mello[a], Michael E. Deagen[a], Joshua Mysona[b], Tzyy-Shyang Lin[a], Juan J. de Pablo[b], Klavs F. Jensen[a], Debra J. Audus[c], Bradley D. Olsen*[a]

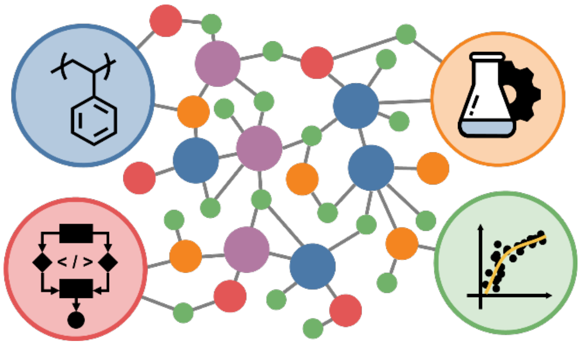
[a]Department of Chemical Engineering Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, United States

[b]Pritzker School of Molecular Engineering, University of Chicago, Chicago, IL 60637, USA.

[c]Materials Science and Engineering Division, National Institute of Standards and Technology, Gaithersburg, Maryland 20899, United States

ABSTRACT:

Polymeric materials are integral components of nearly every aspect of modern life. Today, polymer scientists and engineers devote significant resources to the design and development of these materials to meet growing societal needs. However, developing cheminformatic solutions for polymers has been difficult since they are large stochastic molecules with hierarchical structures spanning multiple length scales from chemical bonds to large molecular assemblies. Here we present the design for a general material data model that underpins the Community Resource for Innovation in Polymer Technology (CRIPT) data ecosystem. Among the key challenges that the data model addresses are the high complexity in defining a polymer structure and the intricacies involved with characterizing material properties. The core design of the data model is graph-based which provides flexibility, robustness, and scalability to support the community-driven mission. This approach to structuring material data provides the key advancements that the community needs to bring cheminformatics to polymer science and accelerate the development of new materials.



INTRODUCTION:

Polymers have transformed the ways we heal, feed, clothe, shelter, and transport humanity,[1–4] and their continual improvement remain a core scientific endeavor.[5–9] Despite the advent of electronic publishing and electronic lab notebooks, the way we record, store, and share scientific data follows largely the same format as it has for many decades.[10–14] The scientific community has recognized the need for improved data infrastructure which has led to the FAIR (findable, accessible, interoperable, and reusable) guiding principles to spur knowledge discovery and innovation by extending the longevity and repurposing of digital research assets.[15,16] The challenge of representing and indexing polymers has hindered the development of data infrastructure, leading to small and disparate data sets or uncatalogued data.[17–20] As a result, most polymer data are scattered across millions of articles and journals in multiple non-interoperable formats.[21–29] The inaccessibility of metadata and data leads to massive inefficiencies and missed opportunities to solve many of our current and future problems with a simple search.[30] Overall, the challenges mentioned above highlight the need for information solutions that make valuable research data discoverable and accessible.

Among the main barriers to developing information solutions for polymers is the fact that they are large stochastic molecules with hierarchical structures spanning multiple length scales from chemical bonds to large molecular assemblies (Figure 1).[11,31,32] To date, there is no single representation that can fully define a polymeric material structure, thus making them hard to index.[18] The stochastic nature of these chemical structures stems from the statistical chemical reactions that are used to produce them. This stochasticity brings about distributions in molecular mass, composition, and topology. A combination of structural descriptors (like a chemical drawing) and distribution information is therefore required to fully define the molecular structure. At larger length scales, microstructures develop from phenomena such as phase segregation and crystallization. These microstructures can be spatially ordered, disordered, or have local patches of order and disorder. The combination of multiple chemical descriptors and length scales of structure makes it extremely difficult to define these materials.

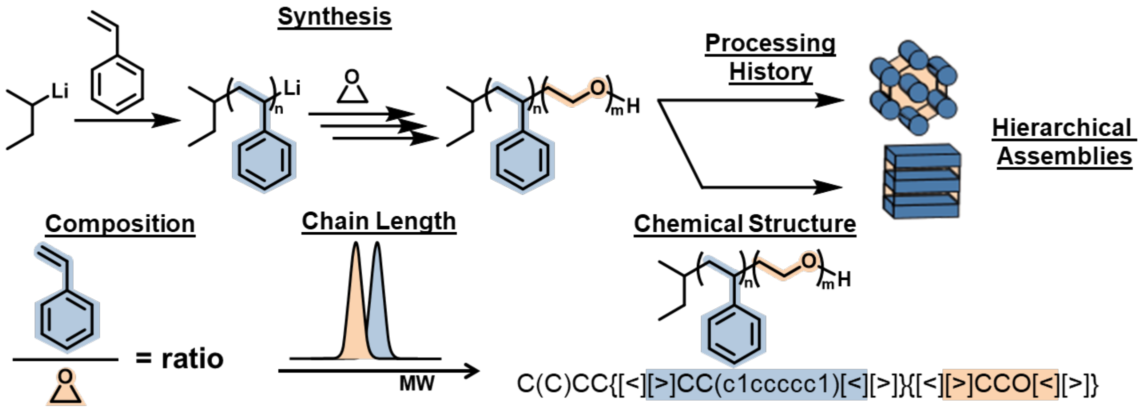


Figure 1: Depiction of the range of data that needs to be captured for an example polymer; poly(styrene – block – ethylene oxide) copolymer.

In practice, defining a polymeric structure and characterizing material properties is even more challenging as polymers have an extremely wide range of properties that often are not easily measured due to physical limitations (e.g., due to poor solubility).[31,33] This leads to variable data availability, often providing only relative information or relying on theoretical models which require expert knowledge to put into context. In many cases, experimentally obtaining the desired information is impossible or intractable, and multiple partial correlative data points must be compiled to provide a surrogate value. Additionally, the processing history under which the material was made can strongly influence microstructure formation and properties. Thus, datasets that do not completely capture the relevant data and metadata will be ineffective at providing reliable information which severely hinders scientific efforts.

With the importance of polymers, there have been several initiatives to capture polymer materials data.[18,22–26,28,29,34,35] Among the key technical innovations needed for success is a data model or blueprint for data organization. Among the simplest and most prevalent schemas for polymers has been the single table schema, where data is structured in a series of rows and columns.[21,28,34] This type of schema is often implemented with polymer names as the key structural descriptor followed by a series of properties. In polymer science, name-based identification has limited capability in specifying a molecular structure, making attributing material properties to structure ambiguous. These types of data sets tend to be small, focused on a limited set of common commercial polymers and properties. The next evolution in database schemas was the migration to single-document schemas.[18,36–38] These databases store data in 'documents' sometimes called objects (i.e., scripted data interchanging formats, typically JavaScript Object Notation (JSON), which encapsulate data and metadata relative to something of interest. For example, PolyDAT[18] focuses its document on a single polymer of interest. These single document style schemas represent a significant improvement from table-based schemas as they introduce flexibility in what data can be stored. However, a major drawback is that the documents need to be organized around an object of interest (e.g., material or reaction of interest). Multi-document, graph, and relational styled databases have emerged as the next evolution in data schemas where data can be referred across documents; thus, reducing the duplication of data and significantly improving the provenance of data. [29,39,40] These data schemas allow for increased flexibility and robustness and contain many features that are essential for storing complex data at a large scale. Another key innovation has been the development of BigSMILES, which extends simplified molecular-input line-entry system (SMILES) a compact line chemical line notation for polymers.[41] BigSMILES provides a human and computer interpretable structural representation that can be used as a key identifier for polymer data.

To address the need for a scalable polymer informatics solution, this work details the Community Resource for Innovation in Polymer Technology (CRIPT) data model. The goal of the CRIPT is to develop a community-driven data ecosystem for polymer science. At the core of CRIPT is a general graph data model that places an emphasis on capturing the metadata and data needed to accurately represent the complexities of polymers material. More specifically, CRIPT's data structure is designed to capture everything from small-molecule and polymer synthesis, material processing, material and reaction properties, material characterization, raw experimental data, and both atomistic and coarse-grained simulations of systems with well-defined chemistries. The data model focuses on providing a highly granular and descriptive design with a strong aversion to ambiguity while seeking to be as comprehensive as possible. CRIPT is driven by FAIR[15,16] and open-source principles[42] to support its community driven mission. The following sections will initially cover the design philosophy and technical aspects that drove the construction of the CRIPT data model. This will be followed by a high-level overview of the main nodes that make up the CRIPT data model and how they are linked to form the graph-based structure. A few examples will be provided to demonstrate how the data model operates. Finally, aspects of the implementation of the data model are discussed. The supporting information contains a detailed discussion for each node and sub-object along with additional examples.

RESULTS AND DISCUSSION

Design Features/Philosophy

CRIPT's data structure is a graph consisting of sets of vertices/nodes, that contain the stored data/metadata, and edges, which store the relationships between data (Figure 2). Every node has a series of attributes and sub-objects. Sub-objects provide a hierarchy for organization and grouping data within a node and attributes are the individual pieces of information that are to be stored. The links between nodes (edges) are achieved with a globally unique and persistent identifier. The presence of the unique identifier of one node in another node signifies an edge between those two nodes in the graph.[29,39,40]

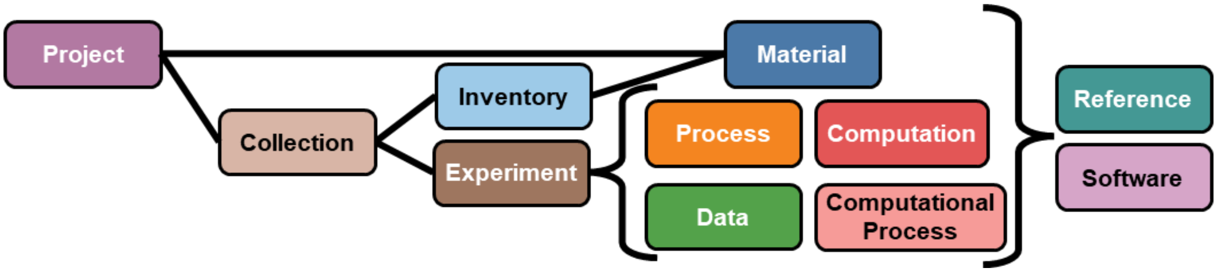


Figure 2: High level view of the hierarchy in nodes that compose CRIPT's graph data structure.

The flexibility of CRIPT stems from the arrangement of nodes in a wide range of configurations, and nodes themselves can be reconfigured with the use of sub-objects. The high level of granularity of CRIPT can be seen in the sub-objects that enable the storage of any type of process, property, and data with explicit specification of context (see sub-object sections in the SI for details). For example, chemical properties can be explicitly attributed to a component in a mixture, or even more granularly, the property can be associated with a fragment of a chemical structure with atom indexing within BigSMILES. CRIPT reduces ambiguity by providing and directing data entry through an ever-growing and curated controlled vocabulary. Data validation is an additional design layer that has been added to minimize ambiguity by ensuring uniformity in the entered data. The combination of these design features set the stage for a robust and general material data model.

The features that make the CRIPT's data model uniquely suited for polymers are the inclusion of process history, support for BigSMILES, and the integration of both experimental and computational data. Capturing process history is a critical detail as physical properties such as elastic modulus, toughness, transparency, etc. depend strongly on processing history. Thus the 'same polymer' can have a wide range of values for a property given its processing conditions. Moreover, the graph architecture supports the need to link characterization data across multiple synthesis and processing steps to enable the definition of complex polymer architectures. CRIPT directly supports BigSMILES as the main structural descriptor for polymers since it is human readable, machine-friendly, and has full support for the wide diversity of polymer structures.[41] BigSMILES in the context of CRIPT can be viewed as defining an ensemble of possible structures for a stochastic polymer where the probability of observing each molecular state is specified with the additional properties captured by the data model (molecular mass, dispersity, etc.). The inclusion of BigSMILES provides an opportunity for polymer structural search as well. CRIPT's data model takes the stance that both data from experiments and simulation should be supported at an equal level, improving cohesion of the research community and the breadth of accessible data.

The design of a data model for an entire community brings about several technical challenges regarding database robustness, performance, maintainability, and cost. CRIPT's graph structure provides robustness by not storing the data relative to a reaction, material, or organizational approach. This enables a single user or multiple different users to enter data which will yield the same representation in the data model. One of the key performance considerations is maintaining fast searching as the size of the database grows, while the data model facilitates rapid graph traversal, indexing by node type, and other advanced database search solutions. Additionally, the reduction of duplicated data through referencing significantly lowers the amount of data that needs to be stored and searched, and persistent identifiers enable reusability. As with any data structure, the design will continue to evolve, and the modularity of the graph structure simplifies maintainability and extensibility as changes are isolated to each individual object, minimizing the cost of reworking/adding improvements (the growth of technical debt).

CRIPT desires to be comprehensive, but it is impractical to store the large and growing amount of material data within CRIPT directly. To navigate this issue, CRIPT's data model embraces federated data storage. Federated data storage is a more attractive alternative to one monolithic server, as the aggregation of all data into a single location is slow, and the database resources are consumed by moving large files around, making these systems much more resource intensive and costly.[43] Support for federated data is realized in CRIPT's data model by focusing on storing key values (such as property data, material identifiers, and processing information) and metadata relevant for discoverability (typically a uniform resource locator (URL)) within the data model while directing non-key information (such as raw data files) to be stored elsewhere. This enables users to store their data on their preferred data services (like Amazon Web Services[49] or university servers). The federated database architecture helps to support the community and a FAIR-driven mission by allowing for decentralization of data.[15,16]

Data Model

CRIPT's data model has two levels of structuring: nodes, which are the primary objects that make up the CRIPT graph structure (nodes will be written in *italics*), and sub-objects which are used to construct sub-structures within the nodes. Several of the nodes serve as organizational tools (*project*, *collection, experiment, inventory*), while the remaining nodes are the core nodes (*material*, *process*, *computation, computational_process*, *data*). Additionally, there is a *reference* node for citing external sources of data and *software* for citing computational tools. A short description and examples of the nodes are shown in Table 1. The following will provide a high-level overview of the nodes; a full detailed explanation of both nodes and sub-objects can be found in the supporting information (SI).

Table 1: CRIPT nodes.

nodes	**short description**	**examples**
organizational nodes		
project	major thrust or research team	sustainable polyurethanes
collection	grouping of experiments	expanding foams, kinetics of ATRP

experiment	grouping of process, data, computations	synthesis of PS-b-PB, Extrusion of PE
inventory	list of material nodes	vinyl monomers, polyolefin library
core nodes		
material	identity and property data	styrene
process	ingredients, quantities, and procedure	anionic polymerization
computation	transformation on data	molecular dynamics, image processing
computational process	virtual process	simulated pyrolysis
data	metadata for raw data or data files	^{1}H NMR
reference	citation to literature	DOI: 10.1038/1781168a0
software	computational tools	LAMMPS
Abbreviations: Atom Transfer Radical Polymerization (ATRP), polystyrene-block-polybutadiene (PS-b-PB), polyethylene (PE), Nuclear Magnetic Resonance (NMR), Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)		

The organizational nodes have a tree like structure with *project* being the root node that represents a major scientific thrust or research group. The *project* node links to one or more *collections* in which a *collection* is roughly equal to a publication or the content of a final project report. A *collection* links to one or more *experiments* and/or *inventories*. An *experiment* in this context refers to the association of data which can be either a physical experiment in the lab or a simulation. The *inventory* node is a way for users to create a list of materials. *Projects*, *collections*, *experiments*, and *inventories* provide organizational tools to help the users who are entering data into the database.

The organizational graph is independent of the core graph structure, with only unidirectional referencing coming from the organization nodes to the core nodes. Structuring the data in this way makes the core data graph invariant to how users organize data in *collection* and *experiment* nodes. Specifically, a *project* node is linked to *material* nodes, and an *experiment* node is linked to *process*, *data*, *computation*, and *computational_process* nodes. *Materials* are only associated with a single *project* node, and those *materials* can only be used within that *project* to avoid issues of data integrity. For example, if a user referenced a *material* node from another *project* and then the *material* was deleted, this would create a broken reference leading to the loss of data integrity. Thus, the reuse of *materials* from other *projects* requires copying *material* nodes into the current *project*. *Process*, *data*, *computation*, and *computational_process* nodes all link to a single *experiment*. Overall, this allows users to organize the *process*, *data*, *computation*, or *computational_process* as they see fit without changing how the data is stored in the database. These design choices seek to minimize the unexpected loss of data integrity without creating a large user or infrastructure burden.

The core graph structure is highly variable depending on the experiment; however, all experimental graphs start with defining a *material* node or set of *material* nodes. A *material* is a collection of the identifiers and properties of a chemical, mixture, or substance. For example, in a typical chemical synthesis (Figure 3), the set of *materials* that are first defined are the ingredients for a *process* node (e.g., chemical reaction). The *process* node contains details about quantities, procedure, process conditions, equipment, etc. for a material transformation. A *process* node may also represent physical transformations (e.g., extrusion) or sample preparation for characterization. An alternating pattern of *material* and *process* nodes serves as the backbone of the experimental graph with *data* or *computation* nodes attached to the *materials* and *processes*. *Data* nodes provide links to sample preparation and to raw or processed experimental data like from nuclear magnetic resonance (NMR), size-exclusion chromatography (SEC), or differential scanning calorimetry (DSC). *Computation* stores information related to the creation or transformation of data. In the example, the *computation* node stores the steps used to calculate material properties derived from the raw data. Multi-step chemical processes, including non-linear diverging and converging processes, are also naturally captured in the core graph structure (see examples 'Diblock Bottlebrush Synthesis' and 'Chemical Reaction with Aliquots' in SI).

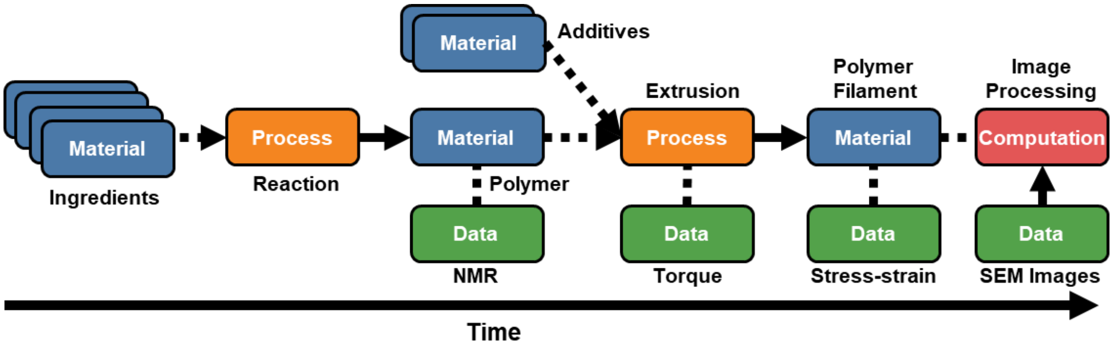


Figure 3: Example of a typical core graph for physical experiments. This example highlights a chemical synthesis (first *process* node) to produce a polymer material followed by an extrusion process (second *process* node). Raw data is recorded at multiple stages of the experiment, and computations are used to convert the scanning electron microscopy (SEM) images into material properties. The arrows between the nodes in the chemical graph show the temporal relationship between the nodes, and the edges without arrow heads imply a 'related to' relationship. Solid lines indicate references that appear directly in the node, where dotted lines indicate references that appear through sub-objects.

For computational experiments, a typical graph starts with a *computation* (Figure 4) to capture the initialization of the computational system and information regarding the set of procedures related to building the molecular structure, initialization of the simulation box, etc. The initial *computation* node will produce *data* nodes that store the configuration of a virtual material. In the example, the initialization *computation* node produces the configuration file for the unequilibrated state of a polymer. *Data* nodes are then passed into further *computations* to transform the virtual material between configurations (e.g., from non-equilibrated state to an equilibrated state). This pattern of alternating *computations* and *data* nodes is a core motif of the simulation graphs. From a virtual material configuration (*data* node), properties can be calculated (*computation* node) and attributed to a polymer (*material* node). The production of the *material* node places simulated material properties in the same position as experimentally determined material properties. The *computational_process* node is used to capture simulated reactions or physical transformations on virtual materials. The *computational_process* node requires both a *material* node and the corresponding configuration file (data node) as the ingredients of the process and will produce a new post-processing configuration file (data node). This new virtual material configuration will lead to a new *material* node as properties are extracted.

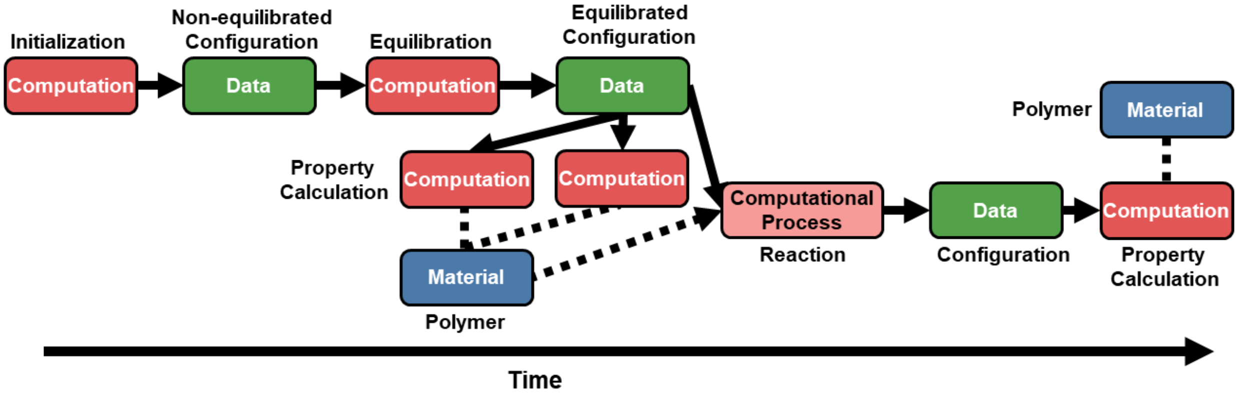


Figure 4: Example of a typical core graph for computational experiments. This example highlights the creation of a polymer system that undergoes a chemical reaction. Material properties are computed for the two materials: pre-chemical reaction and post-chemical reaction.

Data Model Examples

To illustrate the data model, an example graph centered around the organizational nodes is provided (Figure 5). Every new research effort involves creating a *project* node, in this case ‘block copolymer library’. The first part of the project may involve collecting literature information about how to synthesize the targeted block copolymer. During this process, information about the monomers can be recorded in *material* nodes and collected into an *inventory*, ‘vinyl monomers’ for use in experiments. As the project progresses, kinetic experiments were performed to determine the optimal reaction conditions to produce the targeted materials. This set of kinetic experiments is grouped into their own *collection*, ‘ATRP kinetics’. Following successful determination of the optimal reaction conditions, the targeted library block copolymers can be made and grouped into their own *collection*, ‘diblock synthesis’. This example highlights a simple graph for the organizational nodes as there are likely to be many more *collections* and *experiments*.

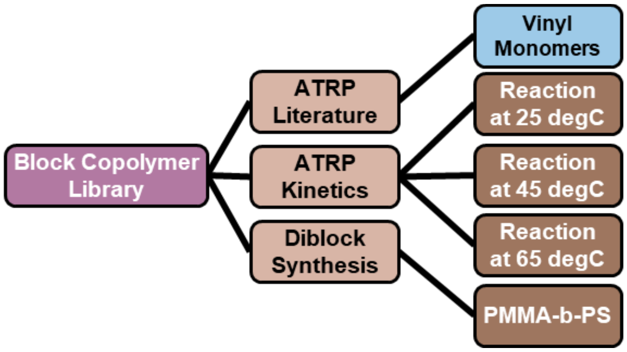


Figure 5: Example graph for the organizational nodes applied to the synthesis of a block copolymer library. The creation of the library involves literature research, kinetics experiments, and then the synthesis of the diblock library. Abbreviations: poly(methyl methacrylate)-block-polystyrene (PMMA-b-PS).

For an example graph of a chemical synthesis, the anionic polymerization of styrene with sec-butyl lithium (secBuLi) in a mixture of tetrahydrofuran (THF) and toluene is illustrated in Figure 6.[44] The chemical synthesis graph is grouped into a single *experiment* node. When defining a new *experiment* for a chemical process, it is recommended to start with the ingredient *material* nodes, followed by a *process* node, product *material* node, and finishing with the characterization *data* nodes. To define the first ingredient in this example, a *material* node for styrene is created by adding identifiers such as SMILES strings, and chemical names. *Material* properties, like density and molecular mass, etc. are also added to aid with calculating reagent amounts. Mixtures such as secBuLi in toluene can be represented by making the two pure material nodes (‘secBuLi’ and ‘Toluene’) followed by making a third material node which is a mixture of the two. With all the ingredients defined, the *process* node for the anionic polymerization is defined by specifying the quantity of each ingredient, experimental procedure, conditions (reaction time, temperature), and reaction properties (yield). In addition to experimental data, this example was inspired by a literature reference, which can be linked directly to the *process* node through a *reference* node. With the *process* defined, the product *material* node ‘Polystyrene’ can now be created by specifying the identity with BigSMILES and a chemical name. Property data such as number average molecular mass and dispersity can be added and the raw data for both the ^{1}H NMR and size exclusion chromatography (SEC) analysis can be attached through the creation of *data* nodes. In this case, the combination of a BigSMILES string and the molecular mass data from SEC provide a full definition of the polymer structure.

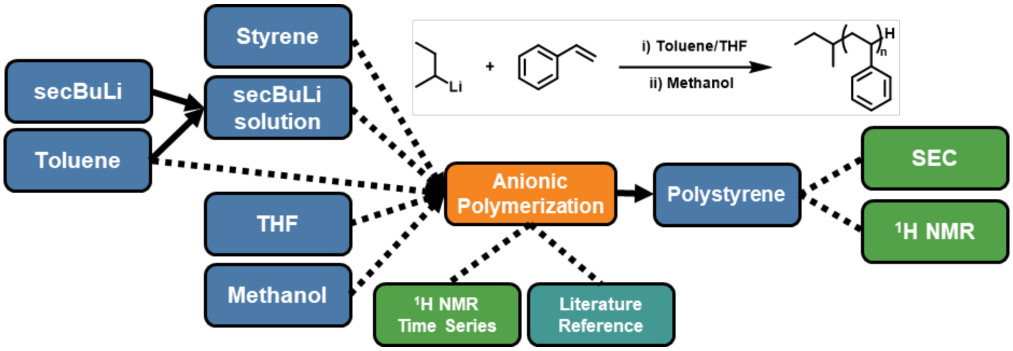


Figure 6: Graph for the chemical synthesis of polystyrene made by the secBuLi initiated anionic polymerization of styrene.

A graph illustrating the investigation of the self-assembling behavior of a block copolymer in thin films (Figure 7) shows the applicability of the data model to polymer characterization. The block copolymer of interest was obtained from a vendor; thus, it was initially characterized with SEC and NMR prior to use. To prepare for the investigation, atomically smooth silicon wafers were cleaned with a plasma treatment and will serve as the substrate for the block copolymer assembly. The first processing approach was to dissolve the block copolymer in acetone and perform a blade coating process. This sample was characterized by AFM (atomic force microscopy) and GISAXS (grazing-incidence small angle x-ray scattering) to determine the microstructure phase and domain spacing. This same sample was then thermally annealed and re-characterized with the same techniques. Following these studies, another film was produced from the original block copolymer sample by dissolving the sample in chlorobenzene and spin coating the solution onto the silicon wafer. The same characterization techniques were once again performed.

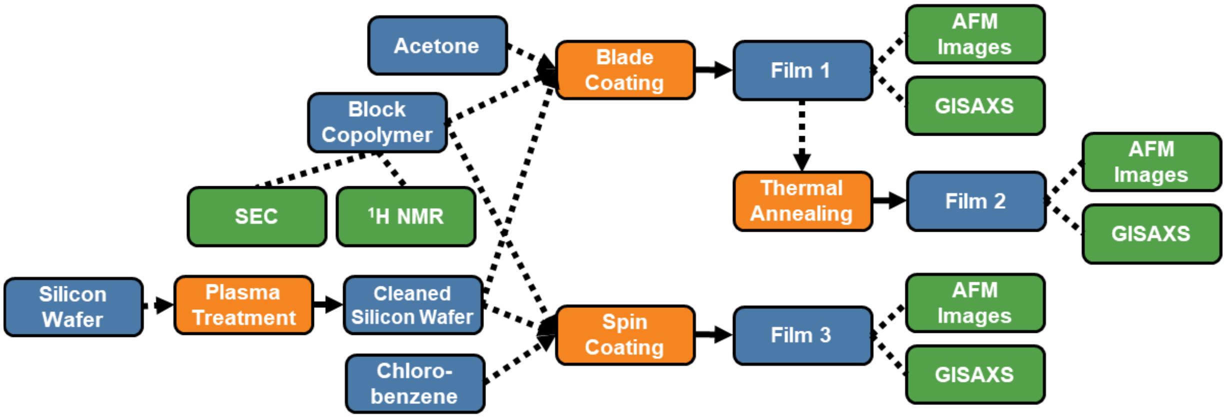


Figure 7: Graph for the characterization of block copolymer films with AFM and GISAXS produced through various processing techniques.

Computational characterization of bulk amorphous polyethylene via molecular dynamics simulations is illustrated in Figure 8.[45] The simulations are conducted using the LAMMPS software with an input file generated by packing polymer chains in a simulation box. After an equilibration procedure with a series of steps to relax, quench, and anneal the system, the radius of gyration, persistence length, and thermal conductivity of the equilibrated polyethylene are measured. For the thermal conductivity measurement an additional heat transfer simulation is required.

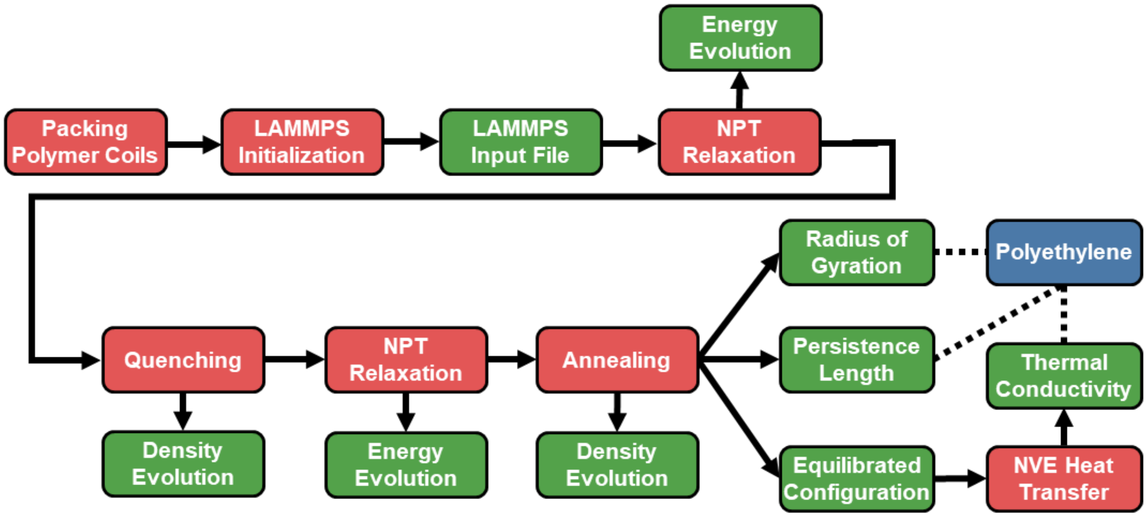


Figure 8: Graph for the thermal conductivity analysis of polyethylene using LAMMPS.

The following example depicts a graph for the extraction of polyolefin material properties from literature for machine learning (Figure 9). The extracted data can be directly stored in the properties section of the *material* nodes and then the material nodes can be organized into sub-data sets with the use of *inventories*. The entire data set can be organized into a single *collection*. Citations back to the literature source can be made on a data point basis with the use of *reference* nodes.

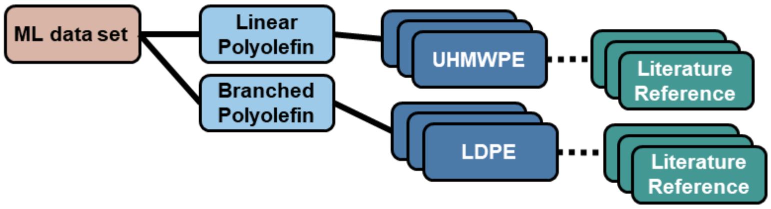


Figure 9: Graph for the creation of a machine learning dataset for the prediction of a material properties (the property data is stored in the *material* node) of linear and branched polyolefins. Abbreviations: ultra-high molecular weight polyethylene (UHMWPE), low density polyethylene (LDPE)

Implementation

To put the data model into practice, a Python[46] software development kit (SDK) has been implemented and made available for download[50]. The purpose of the Python SDK is to streamline the use of the data model, as manually writing data into the data model would be complex, time consuming, and error-prone. The coding style of the Python SDK follows an object-oriented approach where CRIPT nodes and sub-objects are Python Classes, and composition (where a Class instance has one or more other Classes as variables) is used to construct nodes from sub-objects. To handle the referencing between nodes, globally unique and persistent identifiers are autogenerated for each node and used to provide bridges between nodes. In CRIPT the persistent identifiers are URLs as CRIPT is natively web-based (and representational state transfer (REST) compatible), although the use of URLs is not a requirement. Each node can be serialized for storage in any desired format (typically JSON) and transferred across various hardware, software, databases, and programming languages.

The Python SDK provides the opportunity to incorporate various tools to increase data integrity while minimizing the time to enter data. To increase data integrity, data validation layers are included. The first layer is a simple data type check; for example, if the attribute is expected to be a number, the Python SDK will validate that the user is providing a number. In the case where attributes have keys, validation against the officially supported vocabulary is performed. For data that is a quantity (value + unit), the unit dimensionality (e.g., a temperature must have a temperature unit) and value range (e.g., density cannot be negative) will be checked. A few more advanced validations are also present, such as the canonicalization of chemical formulas into Hill system order.[47] This form of validation helps to ensure the uniformity in data and facilitates rapid searching. The Python SDK is written in such a way that additional validation methods can be smoothly added as the software evolves.

To extend the data model for implementation into a full software ecosystem, two additional nodes are included in the data model, *user* and *group*, whose purpose is to provide access control to data. A *user* node is created when an individual joins the CRIPT ecosystem and stores their user information. Among one of the key *user* attributes is ORCID (open researcher and contributor ID) ID which provides a unique and persistent digital identifier back to a specific person.[48] This serves to ensure that all contributions to the database are appropriately attributed to the individual. Additionally, the ORCID ID can be used for login through the ORCID API (application programming interface). A *group* is an organization of multiple *users,* and a *user* can be part of multiple *groups*. The *group* node is where access control/ownership for all data lies, and the *group* node will point to all other nodes in the CRIPT data model. The decision to make *groups* the owner of data was motivated by users tending to change jobs, research groups, and organizations throughout their careers, and data is typically owned by the organization and not the individual. In the simplest case, *group* and *project* will have a one-to-one relationship; and the one-to-one relationship will only be broken when more granular access control is needed (see SI example 'Across Control Within Projects'). The inclusion of these nodes serves to provide a key feature needed to link to user directories common in large organizations. For data that is desired to be shared with the whole community, the 'public' attribute can be set to true to enable the data's inclusion in the public search.

CONCLUSION

This work defines a new graph data model that underpins the CRIPT digital ecosystem. The data model is designed to support data and metadata for polymers from both physical experiments, and both atomistic and coarse-grained simulations of systems with well-defined chemistries. The graph data structure provides flexibility as well as granularity in the data it represents. The connections in the graph provide an intuitive model for the typical material research workflow and allow for high-level information to be swiftly gleaned. Considerations were placed on designing the CRIPT data model to (1) scale for big data while maintaining efficient searching and (2) reduce duplicated data which significantly lowers the amount of data that needs to be stored. This graph-based approach to modeling material data provides the key advancements that the community needs to bring cheminformatics to polymers. Overall, having well-structured data will lead to new innovations and enable the rapid sharing of data and innovations across the scientific community.

SUPPORTING INFORMATION

The main supporting information contains a detailed discussion for each node, sub-object, and additional examples. A second supporting information contains the full data model representation for example found in Figure 6.

AUTHOR INFORMATION

Corresponding Author

*E-mail: bdolsen@mit.edu (B.D.O.)

ORCID

Dylan J. Walsh: 0000-0001-7981-2770

Weizhong Zou: 0000-0002-8369-9229

Ludwig Schneider: 0000-0002-3910-8217

Michael E. Deagen: 0000-0002-8034-0667

Klavs F. Jensen: 0000-0001-7192-580X

Juan J. de Pablo: 0000-0002-3526-516X

Debra J. Audus: 0000-0002-5937-7721

Bradley D. Olsen: 0000-0002-7272-7140

Notes

The authors declare no competing financial interest.

ACKNOWLEDGEMENTS

This work was primarily funded by the National Science Foundation Convergence Accelerator award number 2134795.

REFERENCES

(1) Zhang, X.; Fevre, M.; Jones, G. O.; Waymouth, R. M. Catalysis as an Enabling Science for Sustainable Polymers. *Chem Rev* **2018**, *118* (2), 839–885. https://doi.org/10.1021/acs.chemrev.7b00329.

(2) Matyjaszewski, K. Architecturally Complex Polymers with Controlled Heterogeneity. *Science (1979)* **2011**, *333* (6046), 1104–1105. https://doi.org/10.1126/science.1209660.

(3) Stürzel, M.; Mihan, S.; Mülhaupt, R. From Multisite Polymerization Catalysis to Sustainable Materials and All-Polyolefin Composites. *Chem Rev* **2016**, *116* (3), 1398–1433. https://doi.org/10.1021/acs.chemrev.5b00310.

(4) Walsh, D. J.; Hyatt, M. G.; Miller, S. A.; Guironnet, D. Recent Trends in Catalytic Polymerizations. *ACS Catal* **2019**, *9* (12), 11153–11188. https://doi.org/10.1021/acscatal.9b03226.

(5) Geyer, R.; Jambeck, J. R.; Law, K. L. Production, Use, and Fate of All Plastics Ever Made. *Sci Adv* **2017**, *3* (7), 25–29. https://doi.org/10.1126/sciadv.1700782.

(6) Coates, G. W.; Getzler, Y. D. Y. L. Chemical Recycling to Monomer for an Ideal, Circular Polymer Economy. *Nat Rev Mater* **2020**, *5* (7), 501–516. https://doi.org/10.1038/s41578-020-0190-4.

(7) Schneiderman, D. K.; Hillmyer, M. A. 50th Anniversary Perspective: There Is a Great Future in Sustainable Polymers. *Macromolecules* **2017**, *50* (10), 3733–3749. https://doi.org/10.1021/acs.macromol.7b00293.

(8) Fagnani, D. E.; Tami, J. L.; Copley, G.; Clemons, M. N.; Getzler, Y. D. Y. L.; McNeil, A. J. 100th Anniversary of Macromolecular Science Viewpoint: Redefining Sustainable Polymers. *ACS Macro Lett* **2021**, *10* (1), 41–53. https://doi.org/10.1021/acsmacrolett.0c00789.

(9) Korley, L. T. J.; Epps, T. H.; Helms, B. A.; Ryan, A. J. Toward Polymer Upcycling—Adding Value and Tackling Circularity. *Science (1979)* **2021**, *373* (6550), 66–69. https://doi.org/10.1126/science.abg4503.

(10) Baldi, P. Call for a Public Open Database of All Chemical Reactions. *J Chem Inf Model* **2021**, acs.jcim.1c01140. https://doi.org/10.1021/acs.jcim.1c01140.

(11) Audus, D. J.; De Pablo, J. J. Polymer Informatics: Opportunities and Challenges. **2017**, 1078–1082. https://doi.org/10.1021/acsmacrolett.7b00228.

(12) Kitchin, J. R. Examples of Effective Data Sharing in Scientific Publishing. **2015**. https://doi.org/10.1021/acscatal.5b00538.

(13) Jablonka, K. M.; Patiny, L.; Smit, B. Making the Collective Knowledge of Chemistry Open and Machine Actionable. *Nature Chemistry 2022 14:4* **2022**, *14* (4), 365–376. https://doi.org/10.1038/s41557-022-00910-7.

(14) Moosavi, S. M.; Jablonka, K. M.; Smit, B. The Role of Machine Learning in the Understanding and Design of Materials. *J Am Chem Soc* **2020**, *142* (48), 20273–20287. https://doi.org/10.1021/JACS.0C09105.

(15) Wilkinson, M. D.; Dumontier, M.; Aalbersberg, Ij. J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.-W.; da Silva Santos, L. B.; Bourne, P. E.; Bouwman, J.; Brookes, A. J.; Clark, T.; Crosas, M.; Dillo, I.; Dumon, O.; Edmunds, S.; Evelo, C. T.; Finkers, R.; Gonzalez-Beltran, A.; Gray, A. J. G.; Groth, P.; Goble, C.; Grethe, J. S.; Heringa, J.; ’t Hoen, P. A. C.; Hooft, R.; Kuhn, T.; Kok, R.; Kok, J.; Lusher, S. J.; Martone, M. E.; Mons, A.; Packer, A. L.; Persson, B.; Rocca-Serra, P.; Roos, M.; van Schaik, R.; Sansone, S.-A.; Schultes, E.; Sengstag, T.; Slater, T.; Strawn, G.; Swertz, M. A.; Thompson, M.; van der Lei, J.; van Mulligen, E.; Velterop, J.; Waagmeester, A.; Wittenburg, P.; Wolstencroft, K.; Zhao, J.; Mons, B. The FAIR Guiding Principles for Scientific Data Management and Stewardship. *Sci Data* **2016**, *3* (1), 160018. https://doi.org/10.1038/sdata.2016.18.

(16) Scheffler, M.; Aeschlimann, M.; Albrecht, M.; Bereau, T.; Bungartz, H. J.; Felser, C.; Greiner, M.; Groß, A.; Koch, C. T.; Kremer, K.; Nagel, W. E.; Scheidgen, M.; Wöll, C.; Draxl, C. FAIR Data Enabling New Horizons for Materials Research. *Nature 2022 604:7907* **2022**, *604* (7907), 635–642. https://doi.org/10.1038/s41586-022-04501-x.

(17) Deagen, M. E.; Brinson, L. C.; Vaia, R. A.; Schadler, L. S. The Materials Tetrahedron Has a “Digital Twin.” *MRS Bull* **2022**, *47* (April). https://doi.org/10.1557/s43577-021-00214-0.

(18) Lin, T.; Rebello, N. J.; Beech, H. K.; Wang, Z.; El-Zaatari, B.; Lundberg, D. J.; Johnson, J. A.; Kalow, J. A.; Craig, S. L.; Olsen, B. D. PolyDAT: A Generic Data Schema for Polymer Characterization. *J Chem Inf Model* **2021**, acs.jcim.1c00028. https://doi.org/10.1021/acs.jcim.1c00028.

(19) Xu, P.; Chen, H.; Li, M.; Lu, W. New Opportunity: Machine Learning for Polymer Materials Design and Discovery. *Adv Theory Simul* **2022**, 2100565. https://doi.org/10.1002/ADTS.202100565.

(20) Cencer, M. M.; Moore, J. S.; Assary, R. S. Machine Learning for Polymeric Materials: An Introduction. *Polym Int* **2021**. https://doi.org/10.1002/PI.6345.

(21) Bicerano, J. *Prediction of Polymer Properties*, 3rd ed.; CRC Press, 2002. https://doi.org/10.1201/9780203910115.

(22) Kim, C.; Chandrasekaran, A.; Huan, T. D.; Das, D.; Ramprasad, R. Polymer Genome: A Data-Powered Polymer Informatics Platform for Property Predictions. *The Journal of Physical Chemistry C* **2018**, *122* (31), 17575–17585. https://doi.org/10.1021/acs.jpcc.8b02913.

(23) *Chemical Retrieval on the Web (CROW)*. https://www.polymerdatabase.com/ (accessed 2022-06-24).

(24) *NanoMine*. https://materialsmine.org/nm#/ (accessed 2022-06-24).

(25) CRC Press; Taylor & Francis Group. *CHEMnetBASE - Polymers: a Property Database*. https://poly.chemnetbase.com/faces/polymers/PolymerSearch.xhtml (accessed 2022-06-24).

(26) *Polymer Property Predictor and Database*. http://pppdb.uchicago.edu/ (accessed 2022-06-24).

(27) Brandrup, J.; Immergut, E. H.; Grulke, E. A. *Polymer Handbook*; Wiley New York, 1999.

(28) Mark, J. E. *Physical Properties of Polymers Handbook*; Mark, J. E., Ed.; Springer New York: New York, NY, 2007; Vol. 158. https://doi.org/10.1007/978-0-387-69002-5.

(29) Otsuka, S.; Kuwajima, I.; Hosoya, J.; Xu, Y.; Yamazaki, M. PoLyInfo: Polymer Database for Polymeric Materials Design. *Proceedings - 2011 International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2011* **2011**, 22–29. https://doi.org/10.1109/EIDWT.2011.13.

(30) Himanen, L.; Geurts, A.; Foster, A. S.; Rinke, P. Data-Driven Materials Science: Status, Challenges, and Perspectives. *Advanced Science* **2019**, *6* (21), 1900808. https://doi.org/10.1002/ADVS.201900808.

(31) Hiemenz, P.; Lodge, T. P. *Polymer Chemistry, Second Edition*, 2nd ed.; CRC Press, 2007.

(32) Allcock, H. R. Rational Design and Synthesis of New Polymeric Material. *Science (1979)* **1992**, *255* (5048), 1106–1112. https://doi.org/10.1126/SCIENCE.255.5048.1106.

(33) Barth, H. G.; Mays, J. W. Modern Methods of Polymer Characterization. 561.

(34) Brandrup, J.; Immergut, E. H.; Grulke, E. A. *Polymer Handbook*, 4th ed.; Wiley, 2003.

(35) Park, N.; Manica, M.; Born, J.; Zubarev, D.; Mill, N.; Hedrick, J.; Arrechea, P.; Erdmann, T. An Extensible Software Platform for Accelerating Polymer Discovery through Informatics and Artificial Intelligence Development. *ChemRxiv* **2022**. https://doi.org/10.26434/chemrxiv-2022-811rl.

(36) Kearnes, S. M.; Maser, M. R.; Wleklinski, M.; Kast, A.; Doyle, A. G.; Dreher, S. D.; Hawkins, J. M.; Jensen, K. F.; Coley, C. W. The Open Reaction Database. *J Am Chem Soc* **2021**, *143* (45), 18820–18826. https://doi.org/10.1021/jacs.1c09820.

(37) Adams, N.; Winter, J.; Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML and the World-Wide Web. 8. Polymer Markup Language. *J Chem Inf Model* **2008**, *48* (11), 2118–2128. https://doi.org/10.1021/ci8002123.

(38) Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML, and the Worldwide Web. 1. Basic Principles. *J Chem Inf Comput Sci* **1999**, *39* (6), 928–942. https://doi.org/10.1021/ci990052b.

(39) Citrine Informatics. *GEMD (Graphical Expression of Materials Data)*. https://citrineinformatics.github.io/gemd-docs/ (accessed 2022-01-01).

(40) Brinson, L. C.; Deagen, M.; Chen, W.; McCusker, J.; McGuinness, D. L.; Schadler, L. S.; Palmeri, M.; Ghumman, U.; Lin, A.; Hu, B. Polymer Nanocomposite Data: Curation, Frameworks, Access, and Potential for Discovery and Design. *ACS Macro Lett* **2020**, *9* (8), 1086–1094. https://doi.org/10.1021/acsmacrolett.0c00264.

(41) Lin, T.-S.; Coley, C. W.; Mochigase, H.; Beech, H. K.; Wang, W.; Wang, Z.; Woods, E.; Craig, S. L.; Johnson, J. A.; Kalow, J. A.; Jensen, K. F.; Olsen, B. D. BigSMILES: A Structurally-Based Line Notation for Describing Macromolecules. *ACS Cent Sci* **2019**, *5* (9), 1523–1531. https://doi.org/10.1021/acscentsci.9b00476.

(42) Opensource.org. *Open Source Initiative*. https://opensource.org/osd (accessed 2022-01-01).

(43) Dehghani, Z. *How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh*. https://martinfowler.com/articles/data-monolith-to-mesh.html (accessed 2022-01-01).

(44) Geacintov, C.; Smid, J.; Szwarc, M. Kinetics of Anionic Polymerization of Styrene in Tetrahydrofuran. *J Am Chem Soc* **1962**, *84* (13), 2508–2514. https://doi.org/10.1021/ja00872a012.

(45) Zhang, T.; Luo, T. Role of Chain Morphology and Stiffness in Thermal Conductivity of Amorphous Polymers. *Journal of Physical Chemistry B* **2016**, *120* (4), 803–812. https://doi.org/10.1021/ACS.JPCB.5B09955/SUPPL_FILE/JP5B09955_SI_001.PDF.

(46) Python Software Foundation. *Python*. http://www.python.org (accessed 2021-12-31).

(47) Hill, E. A. ON A SYSTEM OF INDEXING CHEMICAL LITERATURE; ADOPTED BY THE CLASSIFICATION DIVISION OF THE U. S. PATENT OFFICE. *J Am Chem Soc* **1900**, *22* (8), 478–494. https://doi.org/10.1021/ja02046a005.

(48) ORCID. *ORCID*. https://orcid.org/ (accessed 2021-12-31).

(49) Certain commercial products are identified in this paper in order to provide examples. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the product identified is necessarily the best available for the purpose.

(50) Code available at https://github.com/C-Accel-CRIPT/cript.