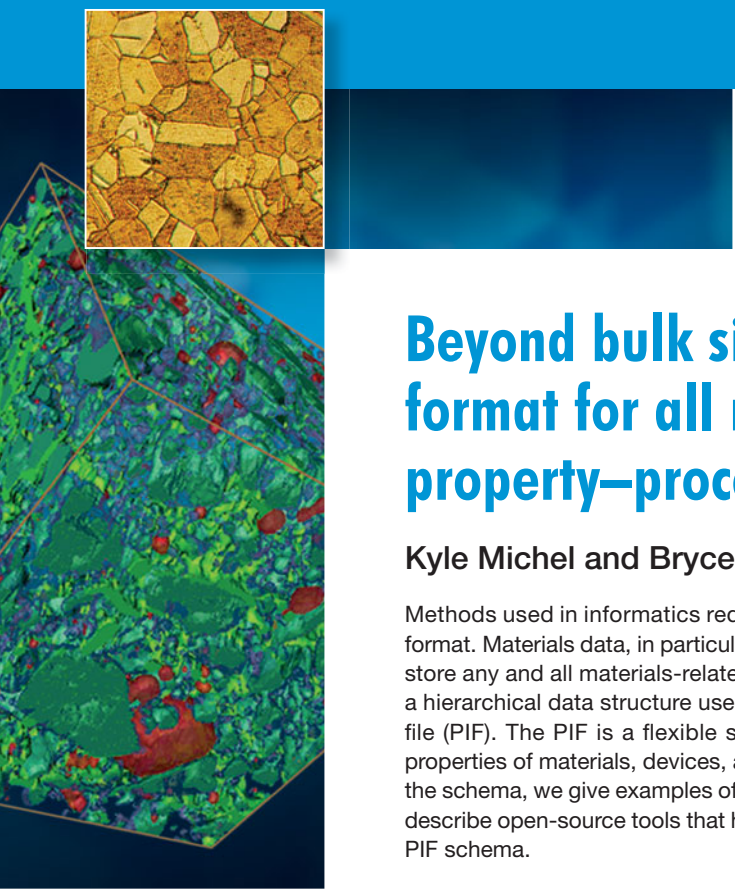


Beyond bulk single crystals: A data format for all materials structure–property–processing relationships

Kyle Michel and Bryce Meredig

Methods used in informatics require input data that are in a machine-readable, structured format. Materials data, in particular, can be exceedingly complex, so defining data formats to store any and all materials-related information is a daunting task. In this article, we discuss a hierarchical data structure used for storing materials data called the physical information file (PIF). The PIF is a flexible schema for storing the structure, processing history, and properties of materials, devices, and physical systems. In addition to a general discussion of the schema, we give examples of its use in representing complex materials systems. We also describe open-source tools that have been developed for building and reading files using the PIF schema.

Introduction

Data science is quickly changing the way in which scientists and engineers make new discoveries.[1–3] Powerful tools in data science are readily available,[4–8] but are only useful when a data set exists to which they can be applied. There is a movement currently in the materials community to build databases of materials, processing, and property relationships.[9–14] Such databases will allow materials scientists to leverage past results into future discoveries, accelerating the rate at which new, advanced materials are developed and deployed.

A central component to any database is the schema used to represent the data that it contains. The materials community has schemas, but few are widely accepted, and most are designed for narrow use. The crystallographic information file,[15] for example, has gained widespread adoption, but is only well suited for the storage of crystallographic data. Recently, this new database movement has inspired the development of more general schemas to represent all information relevant to materials science.[16–19] While these developments are promising, none of these schemas has yet reached widespread use.

In this article, we discuss three important considerations around the design of a materials data schema—generality, flexibility, and ease of structuring. Such a schema must be general enough to represent the wide range of properties and processing relationships that define a material and its behavior, but not so general that information critical to a particular field cannot be modeled. With the sheer breadth of data that it could store, that schema must be flexible to allow for inclusion of data not considered by its developers. Finally, insertion into any schema presents a barrier to most data owners, so any successful schema will make the translation process as simple and low effort as possible.

With these considerations in mind, we discuss the physical information file (PIF), a schema designed to represent a broad array of information related to materials and physical devices, the ways in which they are made, and the properties that they exhibit. We explain the decisions that were made around the design of that schema and the ways in which those decisions address the issues outlined. Examples of physical systems data structured using the PIF schema are presented, including one example of hierarchical microstructure data.

Schema design and the PIF

The decisions made about a schema design directly determine the amount of work that will be required of users putting data into that schema and of users analyzing that data. In this section, we discuss these considerations and the choices that were made in the design of the PIF schema.[20] The PIF schema is intended to store a broad array of information about physical devices and their properties, from parts in a car down to a single monomer in a polymer matrix. In order to handle such a wide range of systems, the choices that are made in the schema

Kyle Michel, Citrine Informatics, USA; kyle@citrine.io
Bryce Meredig, Citrine Informatics, USA; bryce@citrine.io
doi:10.1557/mrs.2016.166

© 2016 Materials Research Society

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

design emphasize generality more than specificity, flexibility more than rigidity, and ease of structuring more than ease of analysis. These topics and the choices made around the PIF schema are discussed in more detail in the following subsections. The exact definition of the PIF schema is not included in the article, but can be accessed online.[20] Instead, examples of data structured in the PIF schema are given in a later section.

Generality of schema

Using a schema that is designed from the beginning with a narrow focus will make the task of storing data outside of the focus area difficult or impossible. For example, a schema that is designed from the beginning to store the properties of elements might not be able to adequately represent molecules, crystals, or polymers, and certainly could not represent, for example, an airplane. As a schema is made more general, a broader array of information can be structured within it. However, if made too general, such a schema could become overwhelmingly detailed and difficult to maintain if additional types are continuously added to it, or, in contrast, it may become too abstract if an effort is made to minimize the size of the schema. As such, it is important to balance generality with specificity.

The PIF schema is designed to store information about any physical system, whether it is a polymer, a battery, or an entire airplane wing, and must therefore be general. In order to prevent an explosion of the schema definition, a key design feature is the recognition of information that is common to all of these systems. These commonalities include some set of identifying information, processing and preparation details, and observed properties of the system. The first two types answer the question "What is this and how was it made?" while the last answers "What are the properties of this thing that was made?"

Top-level systems in the PIF schema can have any number of subsystems, and each of those have additional subsystems. In this way, the object that represents an airplane wing can, through nesting of these subsystems, contain information all the way down to the phases present in an alloy that makes up a single piece of that wing. At each level, standard fields representing the properties of the system can be attached. This allows for complex hierarchies of atomic-level information, through to materials, and up to devices and parts to be fully represented, with the properties of each system structured in a common way.

The PIF schema is not, however, a replacement for all files that store data about physical systems. It will always be true that a schema designed for a particular use will be more specific in the way that it represents that data. For example, it would not be an appropriate use of the PIF schema to store all of the information embedded in a computer-aided design file. However, the PIF provides a common schema into which system–processing–property data can be easily added and analyzed. This common structure facilitates retrieval and analysis across domains.

Flexibility of schema

A rigid schema is one in which no information can be added to a file that is not explicitly stated in the schema definition. Rigid schemas reduce the complexity of analysis since the presence and structure of data can be assumed by the user. However, a rigid schema will quickly prohibit the inclusion of data not specifically addressed in the design. Flexibility in a schema can be addressed in many ways. In the following, we discuss some of the methods used in the PIF schema that allow for flexibility in structuring data.[20]

Using the following as an example, in a completely rigid schema, the bandgap of a material might simply be represented as:

$$\text{bandgap} = 3.4.$$

There are several important consequences of this design: first, the field named "bandgap" must appear in the schema definition; second, the value of that field must be a single, floating point number; and third, a set of implied units, such as eV, must be assumed. This is certainly a valid representation of the data, but it is easy to see that defining a vocabulary to describe every property that one might be concerned about all the way from airplanes down to atoms would become overwhelming. In addition, this schema places work on the user structuring their data to ensure that they have converted values to use the correct units. This same value represented in a less-structured schema could have the form:

$$\begin{aligned}\text{name} &= \text{bandgap},\\ \text{value} &= 3400,\\ \text{units} &= \text{meV}.\end{aligned}$$

While verbose, the user is free to name the property as they see fit and use units that are most convenient to them. Of course, this representation adds work to analysis; someone trying to process all bandgaps in the database might have to rely on a computer application that understands that "band gap," "Band Gap," and "bandgap" all refer to the same property. In addition, comparison of many different values would require that they first be normalized to a common set of units. However, these sorts of tasks are well poised for a computer to solve, and relying on a computer to perform them removes the extra work that a person would otherwise have to perform.

Inclusion of fields and objects that are not a part of a schema definition is another method that can be used to introduce flexibility. Returning to the previous example, in addition to "name," "value," and "units," one might wish to add a field named "uncertainty." While these types of additions are allowed in the PIF schema, such additions are generally not recommended, since it can be difficult to understand the ways in which these unsupported fields relate to other entities. They can, however, be a useful mechanism to store data that could not otherwise be represented; it is almost universally true that individuals, research groups, and organizations will have some internal data that are specific to their field. For example, metadata about a specific measurement would be a good case

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

```
{
    "category": "system.chemical",
    "chemicalFormula": "Li0.0024Ni0.9976O"
    "properties": [
        {
            "name": "Crystallinity",
            "scalars": [ { "value": "Polycrystalline" } ]
        },
        {
            "name": "Resistivity",
            "units": "Ohm*cm",
            "scalars": [
                { "value": "28.8677" },
                { "value": "0.2629" },
                { "value": "0.0466" }
            ],
            "conditions": [
                {
                    "name": "Temperature",
                    "units": "K",
                    "scalars": [
                        { "value": "400" },
                        { "value": "700" },
                        { "value": "1000" }
                    ]
                }
            ]
        },
        {
            "name": "Power factor",
            "units": "W/mK",
            "scalars": [
                { "value": "1.21E-4" },
                { "value": "1.66E-2" },
                { "value": "1.48E-1" }
            ],
            "conditions": [
                {
                    "name": "Temperature",
                    "units": "K",
                    "scalars": [
                        { "value": "400" },
                        { "value": "700" },
                        { "value": "1000" }
                    ]
                }
            ]
        }
    ],
    "preparation": [
        {
            "details": [
                {
                    "name": "Route",
                    "scalars": [ { "value": "Solid state reaction" } ]
                }
            ]
        }
    ],
    "references": [ { "doi": "10.1143/JJAP.38.L1336" } ]
}
```

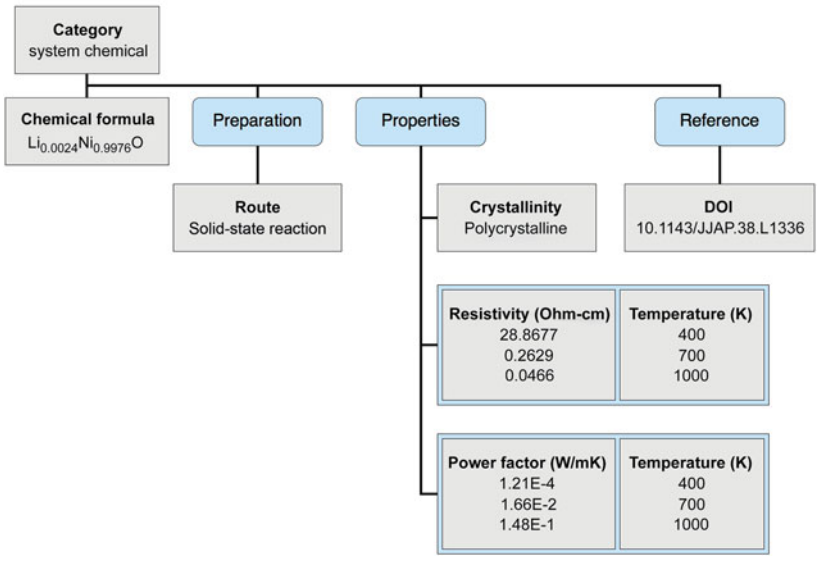



Figure 1. Representation of properties relevant to thermoelectric performance for $Li_{0.0024}Ni_{0.9976}O$ in the physical information file schema. This example shows the resistivity and power factor of the material, both as a function of temperature. In addition, a simple description of the synthesis route is supplied.[21]

for adding new fields since that information is likely particular to that data. A schema definition should not prohibit the incorporation of data that fall into these edge cases, but should instead allow their inclusion and use them to motivate additions to the schema over time.

While some tasks are easy to automate, others are more difficult and should be addressed in the schema design. For example, determining the relationships between entities in an unstructured set of information is incredibly difficult and prone to error when performed completely with computers. Instead, the schema design should make the relationships between entities explicit. In the PIF schema, the hierarchical data structure provides these relationships. For example, a system has a set of properties, each with a name and value, and each property can then have a set of conditions associated with it, each again with a name and value. The hierarchy in this design explicitly defines the relationships between those entities.

Ease of data structuring

The tradeoff between time spent on data structuring versus time spent on analysis has been touched upon already, but it is worth emphasizing the importance that this choice will have on adoption of a schema by a community. A data-driven project will typically follow several steps—collecting, structuring, and depositing of data into a database; retrieval of relevant data to the project; transformation of data into a structure ready for analysis; and data analysis. The goal of the PIF schema is to make the first two steps, structuring and retrieving of data, which are fundamental to all data-driven projects, as simple as possible. Analysis often requires a particular structure to the data such that the transformation step is usually present. The design of the PIF schema takes advantage of this fact and leaves higher level reprocessing of data to the transformation step, rather than trying to design for all possible analyses that could theoretically be applied to the data.

Examples of materials data stored in the PIF schema

This section gives three examples of data in the PIF schema in order to demonstrate the ways in which the previous considerations are incorporated into its definition. These examples illustrate the utility of the schema in representing materials, processing, and properties

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

```
{
    "category": "system.chemical",
    "subSystems": [
        {
            "category": "system.chemical",
            "chemicalFormula": "C14H27N2",
            "names": [
                "C10MI",
                "1-decyl-3-methylimidazolium"
            ],
            "properties": [
                {
                    "name": "Molar mass",
                    "units": "g/mol",
                    "scalars": [ { "value": "223.38" } ]
                }
            ]
        },
        {
            "category": "system.chemical",
            "chemicalFormula": "C2F6NO4S2",
            "names": [
                "TFSI",
                "bis((trifluoromethyl)sulfonyl)imides"
            ],
            "properties": [
                {
                    "name": "Molar mass",
                    "units": "g/mol",
                    "scalars": [ { "value": "280.15" } ]
                }
            ]
        }
    ],
    "properties": [
        {
            "name": "Glass transition temperature",
            "units": "K",
            "scalars": [ { "value": "190.15" } ],
            "references": [
                {
                    "doi": "10.1063/1.2204959",
                    "references": [
                        { "doi": "10.1002/1439-7641(20020215)3:2<161::AID-CPHC161>3.0.CO;2-3" },
                        { "doi": "10.1016/S0040-4039(02)00858-4" }
                    ]
                }
            ]
        },
        {
            "name": "Melting temperature",
            "units": "K",
            "scalars": [ { "value": "244.15" } ],
            "references": [
                {
                    "doi": "10.1063/1.2204959",
                    "references": [ { "doi": "10.1002/1439-7641(20020215)3:2<161::AID-CPHC161>3.0.CO;2-3" } ]
                }
            ]
        },
        {
            "name": "Viscosity",
            "units": "cp",
            "scalars": [ { "value": "152.8" } ],
            "conditions": [
                {
                    "name": "Temperature",
                    "units": "C",
                    "scalars": [ { "value": "20" } ]
                }
            ],
            "references": [
                {
                    "doi": "10.1063/1.2204959",
                    "references": [ { "doi": "10.1016/S0022-0728(03)00343-7" } ]
                }
            ]
        }
    ]
}
```

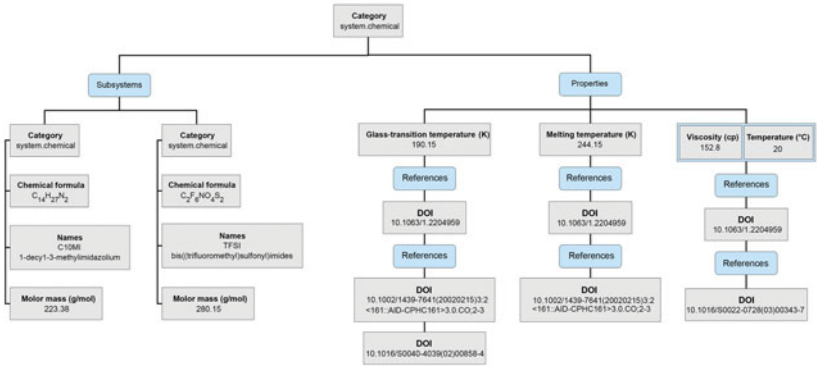



Figure 2. Data about an ionic liquid[22] represented in the physical information file schema. This record gives the molar mass of the two constituent molecules as well as the glass-transition temperature, melting temperature, and viscosity of the ionic liquid.

over a wide variety of use cases, including single and multicomponent systems.

A simple example of information stored in the PIF schema is shown in **Figure 1**. Gaultois et al.[21] obtained data in this record representing several properties of $Li_{0.0024}Ni_{0.9976}O$ relevant to its thermoelectric performance. Several features in this example are worth mentioning. First, the tools used to generate this file automatically add the key "category;" the category is used to mark this record as a chemical system, which informs utilities that read this file about the type of content to expect in the record. Second, three different properties of the material are given—its crystallinity, its resistivity as a function of temperature, and its power factor as a function of temperature. Third, a simple preparation method used to synthesize this material is described.

Zhang et al.[22] obtained data (**Figure 2**) that shows the representation of an ionic liquid in the PIF schema. In this example, the ionic liquid contains two species, which are described in the "subsystems" field. Each subsystem has a chemical formula, several common names, and a molar mass associated with it. The ionic liquid as a whole also contains several properties—the glass-transition temperature, the melting temperature, and the viscosity measured at 20°C. In the same way that system objects can be nested, reference objects can be nested as deeply as needed in order to represent the often complex citation chains attached to materials or measurement.

Finally, **Figure 3** schematically shows the way in which hierarchical system and microstructural information could be structured inside of a PIF record. The outermost system in this record represents a physical part, and the results of a finite element simulation have been stored as properties of that system. The alloy that the part is made from is stored as a subsystem. The S–N (stress–number of cycles) curve for that alloy, a microstructural image, and the results of a grain-growth simulation are stored as properties. Finally, two subsystems of the alloy contain information about its alpha and beta phases. For the alpha phase in particular, an image obtained from scanning tunneling microscopy is stored as well as results of atomistic density functional theory simulations.

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

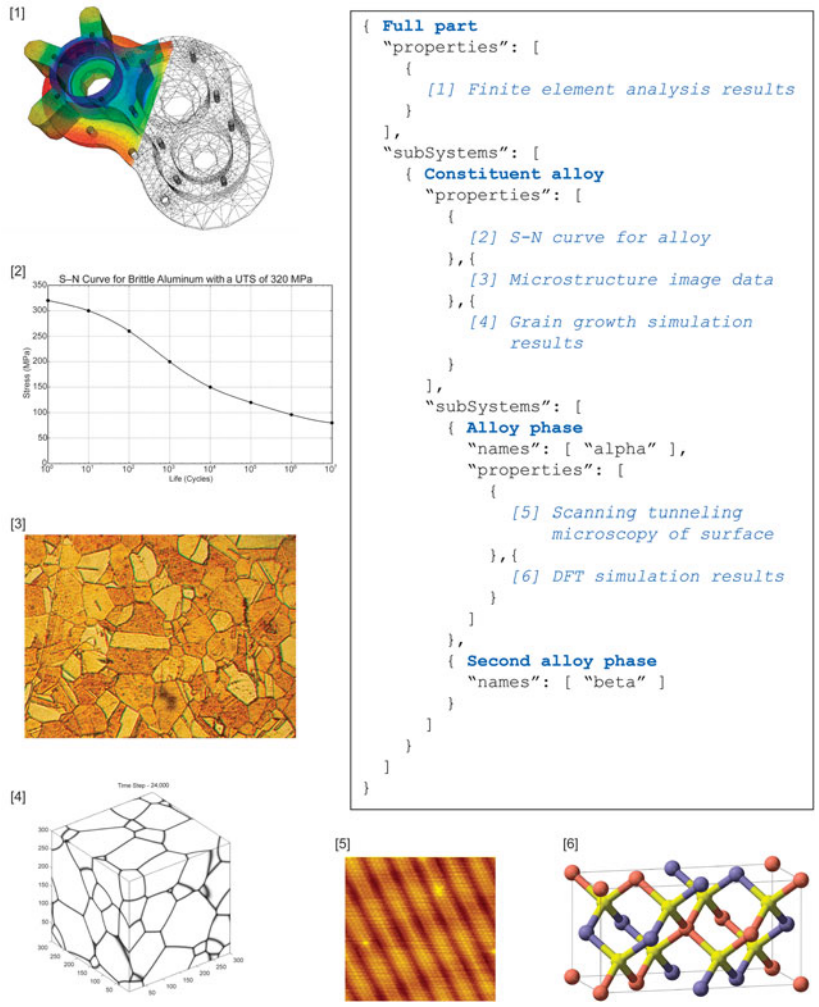


Figure 3. Schematic illustration of hierarchical microstructural data for a material found in a physical part. In this code, blue bolded text labels the systems to improve readability. Blue italicized text holds the place of data shown in the images that surround the figure.[29–34] Micrograph [3] is at 400×. Note: S, stress; N, number of cycles.

The novelty of the PIF schema is that the same data structure that stores information about a thermoelectric material (Figure 1) can be used to represent an ionic liquid (Figure 2), or information about a physical part, its constituent materials, and phases of that material (Figure 3). This singular schema definition allows for easy sharing of data across domains and for the reuse of tools by researchers.

Tools for structuring, retrieval, and analysis

A schema by itself is useful for standardizing data in a common language, but tools around that schema make it more valuable to the community that adopts it. These tools generally fall into three categories—structuring, retrieval, and analysis. Each of these tools can come in the form of a software development kit (SDK) that allows programmatic access to the schema, or a graphical user interface (GUI) that provides graphical access. The importance of tools around a schema should not be minimized. A significant issue that must be addressed for any new data standard is that of incentives; why should a researcher or engineer generating data spend the effort required to get that data into a well-structured format? Tools around a schema can incentivize its adoption since they provide instant value on top of data put into that structure. Particular to the PIF schema, open-source tools around the structuring of data are already available, while those for retrieval and analysis are actively being developed.

Tools for data structuring facilitate the process of organizing information into a schema. Two SDKs are currently being developed around the PIF schema— a Python language toolkit, pypif,[23] and a Java language toolkit, jpif.[24] These toolkits are open source and freely available for download on github.[25] These SDKs provide the ability to generate files in the PIF schema programmatically and to read files already in the PIF schema into python or java objects. It is possible to implement SDKs in other programming languages, or even as a GUI, and such utilities will be developed in the future around the PIF schema.

Tools for retrieval allow for selective extraction of records from a database. While a full-text search on unstructured data is possible, the added information from the use of a schema allows for more control over the records that are returned from a query. The Citrination platform,[26] which hosts more than three million structured materials properties, is an example of a graphical information retrieval system; many other databases exist for topics in materials science and chemistry.[9–14] The Citrination platform, which uses the PIF schema to store records, provides a single repository where information can be compared and analyzed in aggregate, across research groups and industry. Publicly accessible data is hosted on this system free of charge, and tools are provided around that information as a service to those who are generating the data.

Finally, tools for analysis are used to derive insights or make predictions based on some set of input data. Many closed- and open-source packages, such as MATLAB, Scikit-learn, and Apache Spark, are available for analysis of small or large data sets.[4–8,27] These analytics tools are general and can be used on data sets within or beyond materials science and chemistry.

Discussion and outlook

Design of a schema that equally represents all information relevant to materials and their processing and property relationships is a difficult task. While the PIF schema and others have solved

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

in different ways many of the issues around such a data structure, there will always be edge cases for which the schema does not adequately model a particular set of data. With this in mind, the definition of the PIF schema is ever-evolving, incorporating features that allow it to model data outside of its current scope.

Incentives around a schema can drive users to adopt it. In the case of the PIF schema, those tools include the software that was discussed in the previous section. In addition, the Citrination platform[26] provides a single location where users can store those records and have them easily compared and analyzed alongside millions of other records that are stored there. This same platform provides a method of publicizing research results since, as the user base of that system grows, more and more people are interacting with the data there. Finally, as the user base around the schema grows, the rate at which tools are made available grows with the size of the community developing them. While schemas and tools particular to use cases within a research group or company can be useful, these benefits around a single, community-driven standard can incentivize researchers and scientists to spend the effort in structuring their data into this common schema.

Citrine Informatics will continue the development of the PIF schema and the tools used for data structuring and retrieval.[28] However, a strong community around this schema and associated tools will naturally benefit those that use it. With this in mind, the definition of the PIF schema and the source code of the SDKs for working with it will remain open for anyone to contribute to and use. It is our hope that over time, the community around the PIF schema will play a significant role in its development. Individuals with comments or discussion around the PIF schema and SDKs are directed to the websites on which they are hosted.[20,23,24]

References

1. K.M. Tolle, D.S.W. Tansley, A.J.G. Hey, *Proc. IEEE* **99**, 1334 (2011).
2. K. Rajan, *Annu. Rev. Mater. Res.* **45**, 153 (2015).
3. K. Rajan, *Mater. Today* **8**, 38 (2005).
4. MathWorks MATLAB, http://www.mathworks.com/products/matlab.
5. The R Project for Statistical Computing, https://www.r-project.org.
6. Scikit-learn: Machine Learning in Python, http://www.scikit-learn.org.
7. Apache Spark, http://spark.apache.org.
8. Weka, http://www.cs.waikato.ac.nz/ml/weka.
9. A. Frantzen, D. Sanders, J. Scheidtmann, U. Simon, W.F. Maier, *QSAR Comb. Sci.* **24**, 22 (2005).
10. Y. Xu, M. Yamazaki, P. Villars, *Jpn. J. Appl. Phys.* **50**, 11RH02 (2011).
11. "Materials Genome Initiative National Science and Technology Council Committee on Technology Subcommittee on the Materials Genome Initiative" (National Science and Technology Council Committee on Technology Subcommittee on the Materials Genome Initiative, Washington, DC, 2014).
12. A. Jain, S.P. Ong, G. Hautier, W. Chen, W.D. Richards, S. Dacek, S. Cholia, D. Gunter, G. Ceder, K.A. Persson, *APL Mater.* **1**, 011002 (2013).
13. S. Curtarolo, W. Setyawan, S. Wang, J. Xue, K. Yang, R.H. Taylor, L.J. Nelson, G.L.W. Hart, S. Sanvito, M. Buongiorno-Nardelli, N. Mingo, O. Levy, *Comput. Mater. Sci.* **58**, 227 (2012).
14. J.E. Saal, S. Kirklin, M. Aykol, B. Meredig, C. Wolverton, *JOM* **65**, 1501 (2013).
15. S.R. Hall, B. McMahon, Eds., *International Tables for Crystallography Volume G: Definition and Exchange of Crystallographic Data* (Springer, Dordrecht, 2005).
16. J.A. Warren, R.F. Boisvert, *Building the Materials Innovation Infrastructure: Data and Standards* (NISTIR 7898, National Institute of Standards and Technology, 2012).
17. C.H. Ward, J.A. Warren, *Materials Genome Initiative: Materials Data* (NISTIR 8038, National Institute of Standards and Technology, 2015).
18. National Institute of Standards and Technology, "NIST Materials Data Curation System," https://mgi.nist.gov/materials-data-curation-system.
19. P. Huck, A. Jain, D. Gunter, D. Winston, K. Persson, *Comput. Sci. Softw. Eng.* (2015), available at http://arxiv.org/abs/1510.05024.
20. PIF Schema, http://www.citrine.io/pif.
21. M.W. Gaultois, T.D. Sparks, C.K.H. Borg, R. Seshadri, W.D. Bonificio, D.R. Clarke, *Chem. Mater.* **25**, 2911 (2013).
22. S. Zhang, N. Sun, X. He, X. Lu, X. Zhang, *J. Phys. Chem. Ref. Data* **35**, 1475 (2006).
23. Pypif, http://www.citrine.io/pypif.
24. Jpif, http://www.citrine.io/jpif.
25. Github, https://www.github.com.
26. Citrination, https://www.citrination.com.
27. Pandas, http://pandas.pydata.org.
28. Citrine Informatics, http://www.citrine.io.
29. https://commons.wikimedia.org/wiki/File:Elmer-pump-heatequation.png.
30. https://commons.wikimedia.org/wiki/File:BrittleAluminium320MPa_S-N_Curve.svg.
31. https://commons.wikimedia.org/wiki/File:Microstructure_of_rolled_and_annealed_brass;_magnification_400X.jpg.
32. https://commons.wikimedia.org/wiki/File:Grgr3d_small.gif.
33. https://commons.wikimedia.org/wiki/File:Atomic_resolution_Au100.JPG.
34. https://commons.wikimedia.org/wiki/File:Chalcopyrite-unit-cell-3D-balls.png. □



https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press

Publish Alongside Experts in Your Field



Photo by Dennis Schroeder. Courtesy of NREL.

Advance your results. Your research achievements deserve a rigorous review process, the fastest publication time available, and unparalleled reach and impact.

Look for **Feature Issues** on **polymers**, **photonics** and **hard functional materials**....and then publish your best work alongside the experts!

MRS Communications

www.mrs.org/mrc

MRS MATERIALS RESEARCH SOCIETY

CAMBRIDGE UNIVERSITY PRESS

https://doi.org/10.1557/mrs.2016.166 Published online by Cambridge University Press