Computational Materials Science 93 (2014) 178–192

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

A RESTful API for exchanging materials data in the AFLOWLIB.org consortium

Richard H. Taylor [a,b], Frisco Rose [b], Cormac Toher [b], Ohad Levy [b,1], Kesong Yang [c], Marco Buongiorno Nardelli [d,e], Stefano Curtarolo [f,*]

[a] National Institute of Standards and Technology, Gaithersburg, MD 20878, USA
[b] Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA
[c] Department of Nanoengineering, University of California, San Diego, La Jolla, CA 92093, USA
[d] Department of Physics, University of North Texas, Denton, TX, USA
[e] Department of Chemistry, University of North Texas, Denton, TX, USA
[f] Materials Science, Electrical Engineering, Physics and Chemistry, Duke University, Durham, NC 27708, USA

ARTICLE INFO

Article history:
Received 20 March 2014
Received in revised form 5 May 2014
Accepted 10 May 2014
Available online 24 July 2014

Keywords:
High-throughput
Combinatorial materials science
Computer simulations
Materials databases
AFLOWLIB

ABSTRACT

The continued advancement of science depends on shared and reproducible data. In the field of computational materials science and rational materials design this entails the construction of large open databases of materials properties. To this end, an Application Program Interface (API) following REST principles is introduced for the AFLOWLIB.org materials data repositories consortium. AUIDs (Aflowlib Unique IDentifier) and AURLs (Aflowlib Uniform Resource Locator) are assigned to the database resources according to a well-defined protocol described herein, which enables the client to access, through appropriate queries, the desired data for post-processing. This introduces a new level of openness into the AFLOWLIB repository, allowing the community to construct high-level work-flows and tools exploiting its rich data set of calculated structural, thermodynamic, and electronic properties. Furthermore, federating these tools will open the door to collaborative investigations of unprecedented scope that will dramatically accelerate the advancement of computational materials design and development.

1. Introduction

Data-driven materials science has gained considerable traction over the last decade or so. This is due to the confluence of three key factors: (1) Improved computational methods and tools; (2) greater computational power; and (3) heightened awareness of the power of extensive databases in science [1]. The recent Materials Genome Initiative (MGI) [1,2] reflects the recognition that many important social and economic challenges of the 21st century could be solved or mitigated by advanced materials. Computational materials science currently presents the most promising path to the resolution of these challenges.

The first and second factors above are epitomized by *high-throughput* computation of materials properties by *ab initio* methods, which is the foundation of an effective approach to materials design and discovery [3–12]. Recently, the software used to manage the calculation work-flow and perform the analyses have trended toward more public and user-friendly frameworks. The emphasis is increasingly on portability and sharing of tools and data [13–15]. Similar to the effort presented here, the *Materials-Project* [16] has been providing open access to its database of computed materials properties through a RESTful API and a *python* library enabling ad hoc applications [17]. Other examples of online material properties databases include that being implemented by the Engineering Virtual Organization for Cyber Design (EVOCD) [18], which contains a repository of experimental data, materials constants and computational tools for use in Integrated Computational Material Engineering (ICME). The future advance of computational materials science would rely on interoperable and federatable tools and databases as much as on the quantities and types of data being produced.

A principle of high-throughput materials science is that one does not know *a priori* where the value of the data lies for any specific application. Trends and insights are deduced *a posteriori*. This requires efficient interfaces to interrogate available data on various levels. We have developed a simple WEB-based API to greatly improve the accessibility and utility of the AFLOWLIB database [14] to the scientific community. Through it, the client can access

* Corresponding author.
E-mail address: stefano@duke.edu (S. Curtarolo).
[1] On leave from the Physics Department, NRCN, Israel.

http://dx.doi.org/10.1016/j.commatsci.2014.05.014
0927-0256/

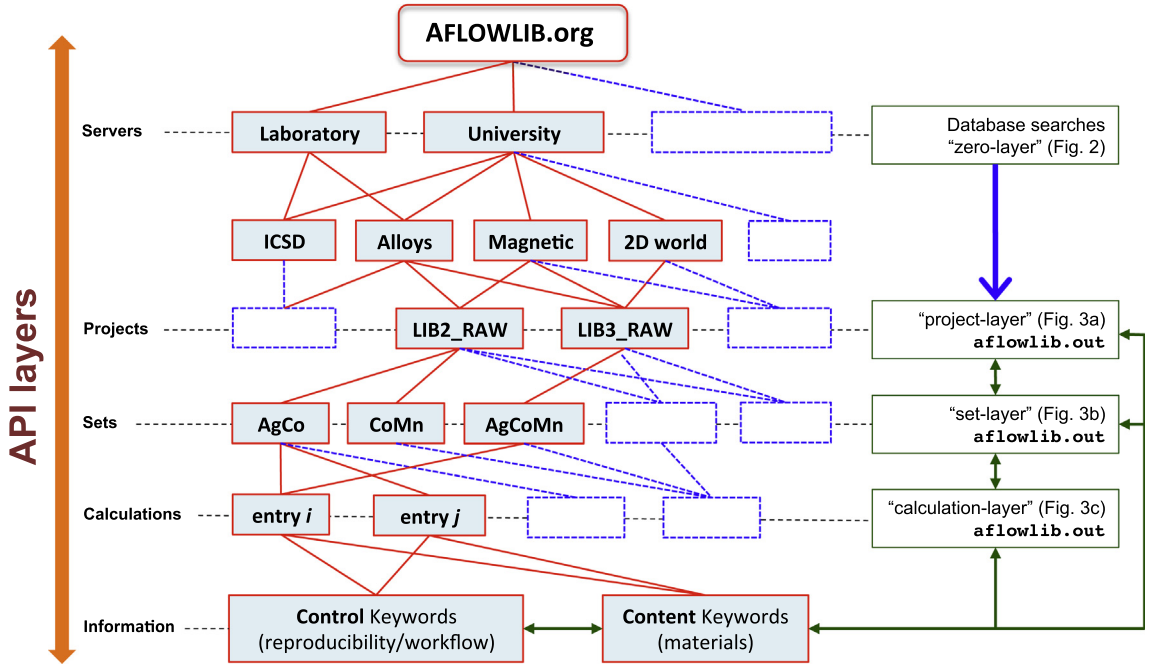

Fig. 1. Schematic structure of the AFLOWLIB consortium.

calculated physical properties (thermodynamic, crystallographic, or mechanical properties), as well as simulation provenance and runtime properties of the included systems. The data may be used directly (e.g., to browse a class of materials with a desired property) or integrated into higher level work-flows. The interface also allows for the sharing of updates of data used in previous published works, e.g., previously calculated alloy phase diagrams [19–31], thus the database can be expanded systematically.

The rest of this paper is organized as follows. The AFLOWLIB libraries are presented in Section 2. A new materials identifier constructed to navigate the libraries is introduced in Section 3. The data provenance and access format schemes are explained in Sections 4 and 5. The access syntax and its various options are described in Section 6. A few examples for the use of the API and two computer scripts are given in Section 7. The strategy for updates is mentioned in Section 8. A brief conclusion is included in Section 9.

2. The AFLOWLIB libraries multi-layered structure

At its core, AFLOWLIB consists of a coordinated set of libraries of first-principles data describing thermodynamic, structural, and other materials properties of alloy systems (Fig. 1). From the top, it is administered through a large SQL database [32] organized in *layers* (reminiscent of other more developed computer interfaces, i.e., IEEE-POSIX [33]). Each layer consists of searchable entries called `aflowlib.out`. The SQL interface is called the *zero-layer* (Fig. 2) because of its structural immanence.

In this *layered* organization, each `aflowlib.out` can be the child of an `aflowlib.out`-parent or the parent of an `aflowlib.out`-child. Each `aflowlib.out` is identified by a name and an address. The first is the AUID (Aflowlib Unique IDentifier – $`auid`), while the second is the AURL (Aflowlib Uniform Resource Locator – $`aurl`). The structure is summarized in Fig. 1.

The current implementation of AFLOWLIB includes three layers that can be navigated using control keywords and absolute paths.

- *Project-layer.* This layer contains information about the project to which the data belongs. For example, for searches in alloys, the project's AURL could be of the type `server:AFLOWDATA/` followed by `LIB1_RAW`, `LIB1_LIB`, `LIB2_RAW`, `LIB2_LIB`, `LIB3_RAW`, or `LIB3_LIB` corresponding to the single element, binary and ternary libraries of post- or pre-processed data respectively. For HTTP access [34], the AURL would be translated from $`aurl=server:/directory/` into $`web=http://server/directory/`. Other translations are considered for future implementations. An example of `aflowlib.out` for the *project-layer* is depicted in Fig. 3(a) (with `server=aflowlib.duke.edu`).
- *Set-layer.* This layer contains information about one or more systems calculated in one or more different configurations (e.g., various structural prototypes, different unit cells required for phonons calculations using the finite difference method, etc.). An example of `aflowlib.out` for the *set-layer* is depicted in Fig. 3(b). To facilitate reproducibility, species making up the $`aurl` might include a subscript or postfix indicating the pseudopotential type. For example, in searches in this layer where the user is interested in the Ag-Ti system, the AURL could be of the type $`aurl=server:AFLOWDATA/LIB2_RAW/AgTi_sv/`. Here the "`_sv`" in `Ti_sv` indicates the "sv" type of pseudopotential in the quantum code used for the calculation, in this case VASP [35]. Other identifiers may be used to indicate pseudopotentials used in, for example, Quantum Espresso (QE) [36] or for potentials coming from other ultrasoft pseudopotential libraries [37].