Journal of
Materials Sciences and Applications

JScholar
Fostering Scholarly Communication

Research Article Open Access

High-Throughput Computing (HTC) Environment for Materials Computational Science Based on the Tianhe Serial Supercomputers

Geng Li, Feifei Li, Guangming Liu, Canqun Yang, Jinghua Feng, Xiangfei Meng*, Xiaoqian Zhu, Xiaodong Jian, Yang Tian, Ziao Jia, Bin Xu, Jian Zhang, Xiaolei Pang, Fuxing Sun, Ren Kai, and Weirong Jiang

National Supercomputer Center in Tianjin, China

***Corresponding authors:** Xiangfei Meng, National Supercomputer Center in Tianjin, China, E-mail: mengxf@nscc-tj.cn

Received Date: April 10, 2021 **Accepted Date:** May 10, 2021 **Published Date:** May 13, 2021

Citation: Geng Li (2021) High-Throughput Computing (HTC) Environment for Materials Computational Science Based on the Tianhe Serial Supercomputers. J Mater sci Appl 5: 1-10

Abstract

The data-driven paradigm is accelerating the discovery of the advanced functional material. Towards the direction, a high-throughput calculation (HTC) environment for materials computational science has been developed based on the Tianhe serial supercomputers. It is a fusion of the cloud environment, supercomputer, and material big-data management. The main features contain the data virtualization, flexible architecture, user-friendly interface, dynamic and automatic workflow, built-in tools, and multiscale simulation. Currently, we have successfully integrated three workflows of material properties in our platform: the calculation of electronic property implemented by Vienna Ab Initio Simulation Package (VASP), the simulation of the interface diffusion implemented by LAMMPS, and the simulation of the dynamics of fluid performed with Open FOAM. In the future, we are going to integrate more multi- and cross-scale material high-throughput computational workflows. The material property repository will also be constructed to be open for material researchers.

Keywords: High-Throughput Calculation; Automatic Workflow; Materials Simulation; Supercomputing

Introduction

Today, advanced materials hold the key to tackling some of our most pressing societal challenges, such as global climate change and our future energy supply [1]. Computational science has now emerged as a new paradigm straddling experiments and theory. Especially, using big data to accelerate the materials discovery has become the first innovation gradually [2,3]. The traditional "trial-and-error" experimental method and "one-by-one" computational method for the materials design are hard to accumulate the massive data sets in the short time. A high-throughput computation infrastructure based on the superconducting center has been proposed by the Materials Genome Initiative [4], which aims to efficiently harness the use of all available resources, continually submit jobs over a long period of time. The advantage to the high-throughput computation is to employ large amounts of computational power to increase the sheer number of calculations performed without significantly increasing the computational cost per calculation [5,6].

Several efforts has been devoted to build the HTC environment by computer and material researchers [7,13]. For instance, in America, the Materials Project is implmented on the National Energy Research Supercomputing Center [14,15], and a joint collaboration between the Cray Supercomputer Company and Duke University develop the Automatic-FLOW for Materials Discovery (Aflow) [16,17]. In Europe, the largest repository for computational materials science worldwide, the Novel Materials Discovery (NOMAD) Repository, is built based on the joint network of the Barcelona Supercomputing Center in Spain, CSC-IT Center for Science in Finland, the Leibniz Supercomputing Centre and the Max Planck Computing and Data Facility in Germany [2,18]. Since 2016, the govement of China has proposed the Materials Genome Engineering Program, which aims to double the speed and save the cost with which scientists discover, develop, and manufacture new materials.

Funded by the program, the National SuperComputer Center in Tianjin cooperating with nine complementary research groups in the materials science established the HTC environment for computational materials science based on the Tianhe serial supercomputers. The success of the practices provide some experiences and challenages that the realization of scientific HTC environment requires the fusion and improvement of computational power, big-data management, theoretical methods, and software [19,22].

In this paper, the practice in constructing the HTC environment for the material science is introduced. Firstly, the feature and architecture of the infrastructure is presented, the typical material computational automatic workflow is discussed, then the big-data management systems is illustrated. And the future works is prospected finaly.

Overview of the HTC Environment

In the 1960s, the development and advancement of density functional theory (DFT), semi-classical thermodynamics theory and finite element method constructed a theoretical framework for accurately predicting the electronic-scale, mesoscopic and macroscale properties of materials [13]. The rise of the computer science technology and the excellent algorithms and languages lay the foundation to realize these theories by various practical codes and software, such as VASP, LAMMPS, Gibbs, ANSYS. Despite significant improvements in robustness and user-friendliness of first-principles codes, limited by the computational resources and traditional work mode, it is still manual and one-by-one to submit the job to calculate and simulate the single material structure, simple properties and special physical condition by experienced material scientists in the laboratory. This traditional method is time consuming and labor intensive, and requires serial steps and actions:

i. Scientists need to learn the knowledge of the physics and software.

ii. Then design the logic calculation process.

iii. Prepare the input files, write the scripts, search, build and optimize structures.

iv. Submit and manage jobs to the remote supercomputers untimely and repeatedly.

v. Require the human intervention to fix the parameters when trivial results and running errors appear.

vi. Finally write post-processing codes and physically analyze to acquire the reliable and graphical results.

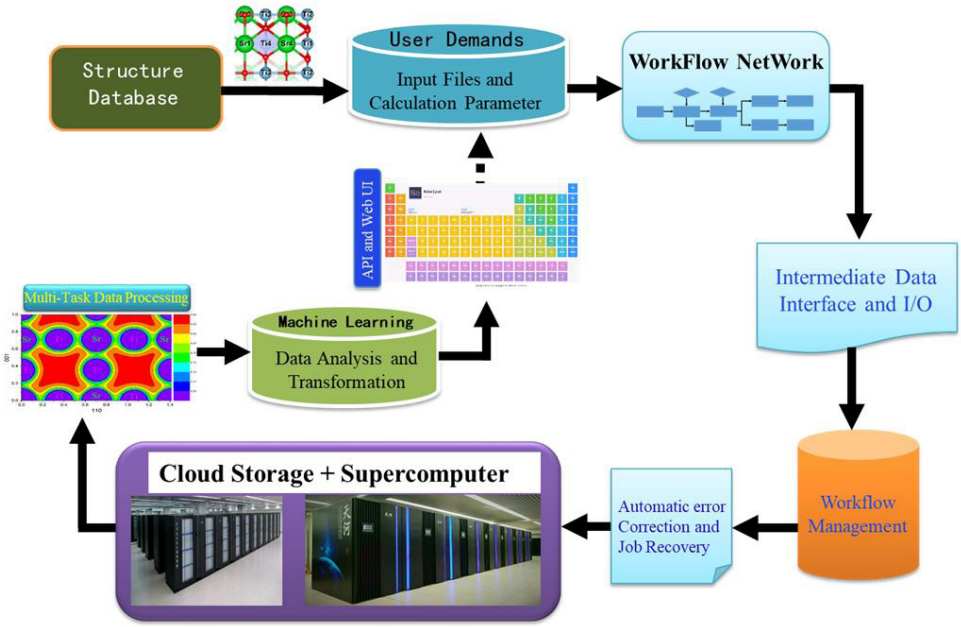

Figure 1: Work process of our HTC environment for material computational science

To overcome these obstacles and simplify execution steps, we developed the HTC Environment for materials computational science based on the Tianhe serial supercomputer. Its main goal is to create, collect, store and cleanse computational materials science data which are computed by the most important materials-science codes, such as the first principle theory and finite element method, automatically executed and analyzed for the display on the website. The operation procedure only requires user to specify what properties and which materials they want to calculate, edit complex input parameters and perform a one-click submit on the web interface. The preparation of input files, the copying of files and directories, the execution of tasks, and the processing and obtain of physical results will be automatically realized in the backend.

Figure 1 displays the complete running of the workflow. The automatic workflow for the high-throughput material calculation system is firstly to construct fixed templates with edited input parameters. According to the requirement of the scientific research, it can automatically prepare input files and design calculation parameters complementary with importing the structural data and potential file from the specific database [23,24]. Based on the orientation of the workflow, users submit some tasks to different computing software, and realize the concurrent computing of different material properties. During the job running, it is convenient for users to manage these workflows, control tasks, perform the intermediate data processing, and examine the running status. At the same time, in order to ensure large-scale and long-term calculation, automatic error correction and recovery continuation can also be realized. After the calculation completed, multi-task data can be processed uniformly, and physical results can be analyzed and transformed through machine learning and other methods. The multi-type data, such as images, binary file and text will be classified. Adopting the high-speed storage and access, a searchable materials database is organized to interact with the web users through the API and Web UI. The whole process requires the fusion of the supercomputing system and cloud storage platform as hardware supports. At the same time, it needs to establish workflow control system, automatic error correction system, data processing storage system, and need to write the python inter-transformation interface for the data exchange between different software and data processing conversion, web display, etc.

Architecture of the Infrastructure

This computational infrastructure is created to discover a new and better materials using high throughput computations. Considering the realization of the workflow and ensure the safety of the HTC environment, the hardware architecture of the infrastructure is separated into five parts, as shown in the Figure 2.

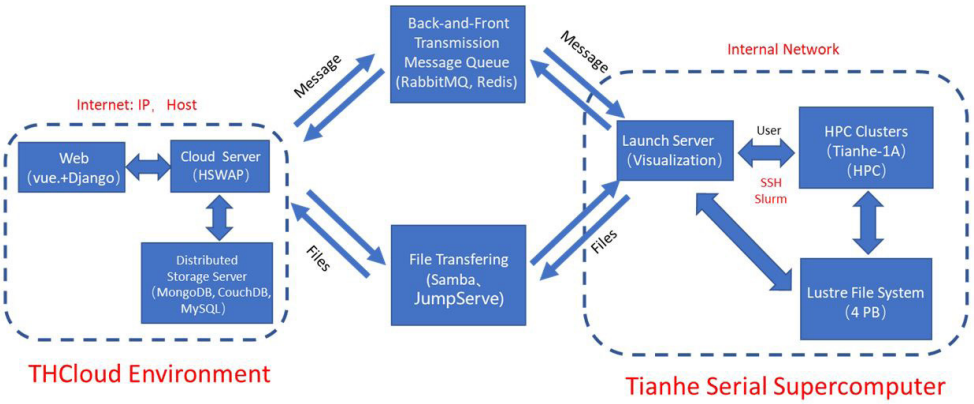

Figure 2: Architecture of the infrastructure. It is composed of three parts: THCloud Environment, intermediate server, and Tianhe serial supercomputer

Hardware Components with their Services and Functions

Cloud Server (Web server): It is composed of several virtual machines. The hostname/IP of the virtual machine is assigned to deploy the software and services. We utilize the popular Vue web framework to build the frontend interface. Vue is a progressive framework for building user interfaces, and it is easy to learn and design the web function because of its various existing components. The backend developed by the Django framework are deployed to connect the users and the backend service. The Django framework has a default interface to a MySQL database and it is very easy to build forms to manage the user information and the mapping relationship between the registered user and applications. The user and software information is managed by the MySQL. On the web frontend, we use centralized user management, a single sign-on access to all backend services.

Storage Servers: It connects the web server with the Lustre file system by the Samba service. It stores the post-processed physical properties of calculated materials structures. The data that users upload and download will be accessed and stored in the database. The database management engines

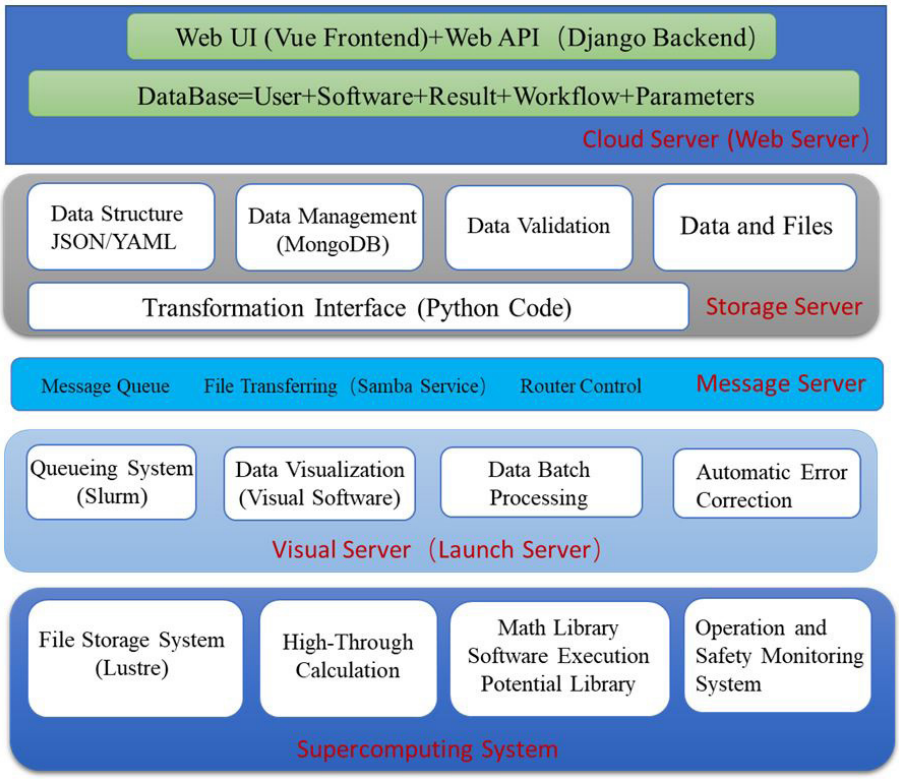

Figure 3: Services and functions for each parts of the infrastructure

MongoDB, MySQL and CouchDB will be installed on the Ceph distributed storage, which ensure the efficient query and the flexibility of the data storage capacity. The input parameters and output physical properties of different material structures will be read and written with the uniform key values in the JavaScript Object Notation (JSON) document that is easy to be managed and handled by the Python code and MongoDB. We mainly adopt MongoDB as the database backend to build the our material repository.

A feature of MongoDB is that both the query language and the native data model are JSON, which is the standard data format for modern web applications and easily represented and manipulated as native Python dicts. CouchDB is a built-in database in the workflow management engine, which is in charge of storing the workflows and its parameters.

Message server: It connects the Tianhe Cloud (THCloud) environment with the Tianhe serial supercomputer. The cross-domain strategy separates physically the web internet with the Tianhe internal network. Serval security strategies, such as firewall, unified authentication, are adopted to prevent the illegal access, intrusions, and attacks. The RabbitMQ and Jump server service are deployed on the message server. It realizes the front-and-back message transmission. Three pair message queues guarantee the stability of the workflow. In addition, the large-size file transferring management system-the Samba service-is also controlled by the message server.

Launch Server (Visual Server): When executing jobs on a large computing cluster, job requests must often pass through a queueing system. Our workflow engine sends messages and files to the launch server. It parses commands, prepares the job and submit the job to the queueing system. Tianhe serial supercomputer uses the Slurm conductor. The developing automatic error correction function can be operated to re-run the single job by the launch server. Another function for the launch server is to process batch resulting data and visualize the data by the installed visual tools. Many post-processed codes will be executed for the output files. For example, the calculation of the alloy including hundreds of thousands of atoms by LAMMPS produces about dozens of GB trajectory file which will be visualized on the visual server instead of the Web Graphics Library (WebGL). The visual materials images will be transformed to the web interface piece by piece.

Supercomputing System: The supercomputing system includes the computing nodes and file storage system. The high-throughput jobs will be running on the computing nodes and pass the raw result data to the file storage system. Our file system is the Lustre with the capacity of 4 PB. The operation and safety monitoring system is deployed to check the running status of the high-throughput jobs and the supercomputer. The statistics of the job status will be pased to the web interface. The different versions of material computing softwares with their dependent math library will be installed in advance. We also develop an potential library which stores a larage amount of potential files required by the input of softwares. The potential library provides a interface to download and transfer by user calculating demands.

The function of every part described above is illustrated in the Fig. 3. Currently, based on the work mode, we have realized three workflows of materials properties in practice. The atomic scale calculation of the electronic property is performed by using the first-principle software VASP [25,26]. The key in realization of the workflow is to automatically generate the band path of the crystal cell, which is solved by using the Seek-Path code [27,28]. Due to the small size of the data generated by the atomic scale calculation, we adopt the WebGL to display the material structure (e.g. POSCAR) and band structure.

The another practical example is to simulate the interface diffusion by the molecular dynamics theory implemented by the LAMMPS [29,30]. The system containing ten thousands of the Mo-Ag alloy has been tested. The logical workflow simulates the annealing process including four steps: relaxing, heating, holding, and cooling. The large-size trajectory file has been extracted and processed by the visualization software OVITIO. All output files can be download by the netdisc on the web windows.

The typical macroscopic material software Open FOAM is also integrated to simulate the dynamics of fluid based on the Navier-Stokes (N-S) equations [31]. Unfortunately, we can't find a better way to automatically generate the mesh file because the surface of the artifact is very complex and we need to find some software and programs like Hub-mesh to generate mesh file in the macroscopic workflow. Users have to upload the polyMesh directory.

Currently the focus is on atomic scale calculations of thermodynamic and electronic properties using density functional theory, but the methodology is applicable to many different length scales, properties, and methods.

Features

Summary of the Features: In short, we summarize the six features of our HTC environment for the material computational science.

Data Virtualization: All operations for users, including the display of the results, the check of the running status, the creation and submission of jobs, can be finished on the graphical web interface. The resulting data can be automatic visualized by built-in visual tools and the intuitive graphical results display on the web.

Flexible Architecture: Our designing architecture of the HTC environment is flexible and expandable for the different material application and calculating scenes. For instance, the capacity of the storage system is expandable according to the amount of data.

User-Friendly Interface: Users can directly interact with the bottom supercomputing resources on our website instead of the command-line Linux windows. It is easy for beginners to learn and use.

Dynamic and Automatic Workflow: The physical workflow designed by users can be convenient to integrate in our platform. The created workflow can also be edited according to user demands.

Built-in Tools: A large number of post-processed codes and visualization software are integrated in the backend. They are attached with lots of interface programs. User can directly use them.

Multiscale: The platform supports to employ different software and methods to calculate different physical properties

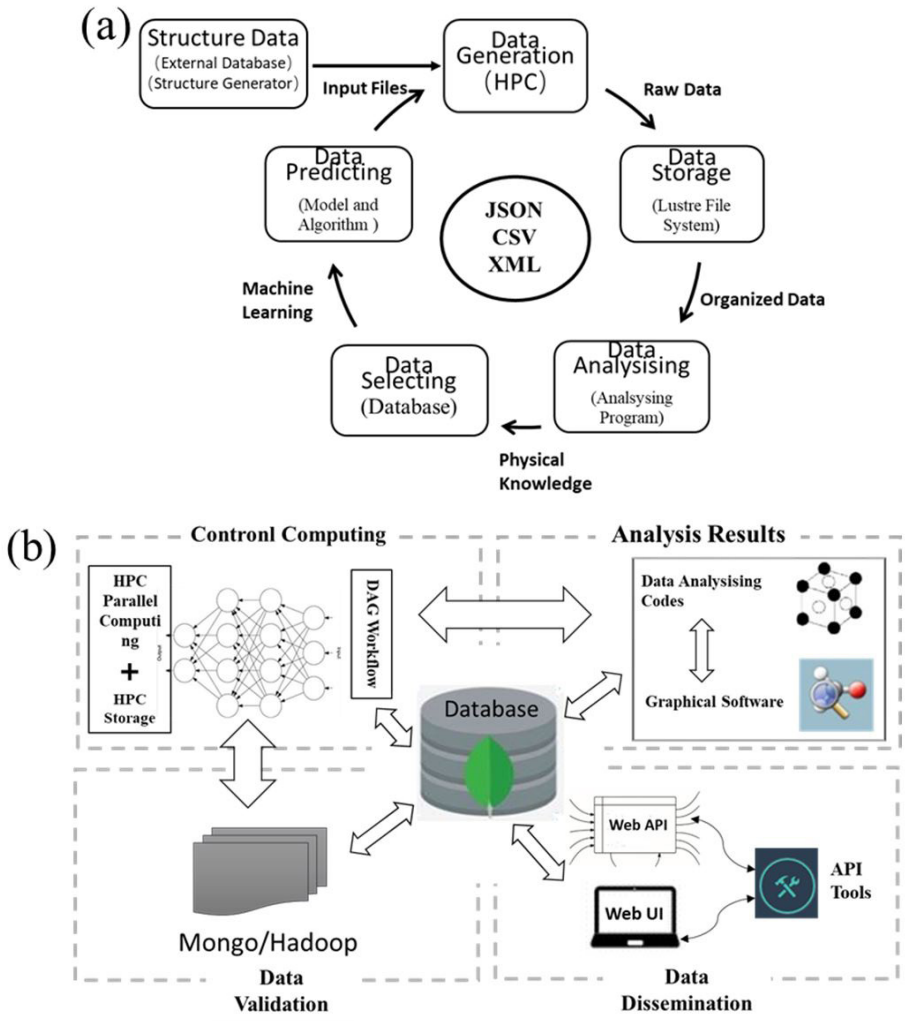

Figure 4: (a) The oriented flow of the material data in the supercomputer.
(b) The function of the core database.

of multi-scale material structure. The cross-scale calculation is ongoing.

Scientific Automatic Workflow

The wide availability of high performance computing (HPC) systems, Grids and Clouds, allows scientists and engineers to implement more and more complex applications to access and process large data repositories and run scientific experiments in silico on distributed computing platforms. A workflow is a well-defined, and possibly repeatable, pattern or systematic organization of activities designed to achieve a certain transformation of data. Many-task materials computing workflows are increasingly using environments due to their need for large computation and storage resources. HPC environments present challenges for running both the data stores and the associated calculation workflows because these environments are originally designed to serve the needs of large MPI applications that run for predictable times and do all I/O to disk [32].

We developed a workflow engine to define, create, control and manage the task execution, named as HSWAP. The logical physical process pre-defined by users can be convenient to integrate into it. Once created, the workflow model can be stored in the CouchDB and then repeatedly multiplexed. Essentially, a workflow created by HSWAP is a directed acyclic graph (DAG) of individual jobs, where the edges represent control dependencies and the nodes denote the execution of the jobs. The node sends messages to call shell scripts, transfer files, write/delete files, or call other python functions. The edge conducts to store data, pass data, or dynamically modify the input of its children node. The existing software routines, datasets, and services in complex compositions have been written in the fixed format [33].

We have developed a web interface of the workflow. Each computing function corresponds to a pre-defined workflow template in the HSWAP engine. The user creates a new workflow on the frontend, which is a clone based on the template. The workflow contains the logical execution and the parameters required for the calculation. The saved calculation parameters can be viewed and changed through the parameter form. The user can independently control the execution of each workflow, and obtain the calculated results after the complete of the calculation, and visually analyze the results using a visualization tool.

Big-data Management

HPC calculations produce a large amount of heterogeneous data. The process of generating material data in the supercomputer is exhibited in Figure 4(a). It contains the structure data input, data generation by HPC, data storage, data analysis, data selecting, and data predicting [34,35]. The data transferring is in the basis of JSON format.

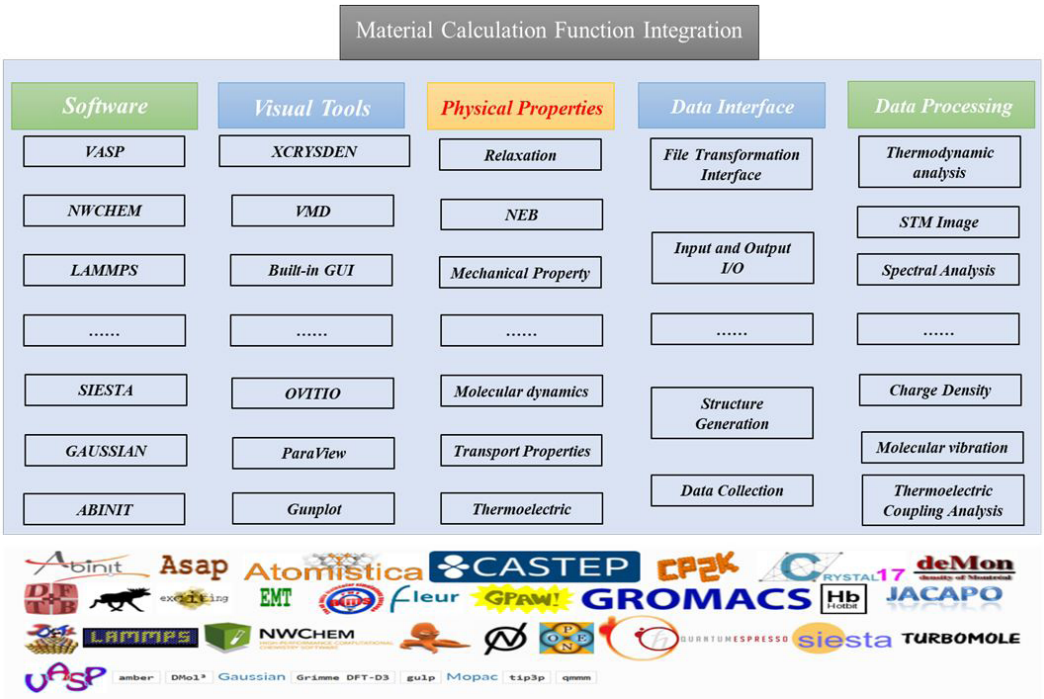

Figure 5: Five parts of material calculation function integration

On one hand, organized files containing input parameters and final results need to be automatically and permanently stored for future reference and analysis, which is stored in a distributed multi-layered storage system connected to scalable computing capacity. They are managed by the MongoDB database. The function of the core database is shown in the Figure 4(b), which includes control computing, results analysis, data dissemination, and data validation. We will create data repositories for easy access and reuse of research data. The data template is a JSON document pre-defined with physical keywords. The data after processed, analyzed, selected, verified in each calculation will be saved in the fixed template. On the other hand, much of the raw data directly generated by the software is required only temporarily, which is stored in the Lustre file system with a big capacity. They will be deleted in regular.

Integration with the Material Application Interface

As shown in the Figure 5, the material calculation can be divided into five parts, including physical properties, data interface, visual tools, software and data processing. The data transferring interfaces will be built to attach them together. The workflow realizes the execution of the five parts in order. The calculated physical properties, such as mechanical property, transport properties, optic properties, identify the DAG of the workflow. The data interface writen with python code, includes file transformation interface, input and ouput I/O, structure generation, etc. Visual tools are some existing softwares running on Linux system, such as XCRYSDEN, Gunplot, VMD. Material computing softwares are composed of commercial and open-source codes like VASP, LAMMPS, Quantum Expresso. The raw resulting data generated by material computing softwares will be analyzed and processed to become physical quantities or images. The common data processing is thermodynamic analysis, charge density processing, molecular vibration analysis, etc.

Summary

The development of the HTC environment for materials computational science are summarized based on the Tianhe serial supercomputers. HTC environment is composed of five parts with their corresponding services and functions: cloud server (web server), storage server, message queue server, launch server (visual server), and supercomputing system. The core of realizing HTC is to design the efficient scientific workflow according to the material knowledge. A workflow management engine is in charge of the creation, execution, and management of tasks automatically and dynamically. Various python interface codes have been developed to process the raw data. The processed result data in each workflow will be saved in the templated JSON document. We expect to make a progress in the multi-scale calculation, extreme-scale computing, and machine learning to predict advanced functional materials.

Acknowledgment

This research was supported by the National Key Research and Development Program of China under Grant No. 2018YFB0204305 and 2018YFB0703900.

References

1. J Hachmann, R, Olivares Amaya, S Atahan-Evrenk, C Amador Bedolla, RS Sanchez Carrera, et al. (2011) The Harvard clean energy project: large-scale computational screening and design of organic photovoltaics on the world community grid, J Phys Chem Lett 2: 2241-51.

2. C Draxl, M Scheffler, NOMAD (2018) The FAIR Concept for Big-Data-Driven Materials Science, Invited Review for MRS Bull 43: 676-82.

3. JP Correa Baena, K Hippalgaonkar, J van Duren, S Jaffer, VR Chandrasekhar, et al. (2018) Accelerating materials development via automation, machine learning, and high-performance computing, Joule 2: 1410-20.

4. Materials Genome Initiative for Global Competitiveness (2011).

5. B Blaiszik, K Chard, J Pruyne, R Ananthakrishnan, S Tuecke, et al. (2016) The materials data facility: data services to advance materials science research, Jom. 68: 204-52.

6. D Gunter, S Cholia, A Jain, M Kocher, K Persson, et al. (2012) Community accessible datastore of high-throughput calculations: experiences from the 5Materials Project, In 2012 SC Companion: High Performance Computing, Networking Storage and Analysis IEEE 1244-51.

7. G Pizzi, A Cepellotti, R Sabatini, N Marzari, B Kozinsky, AiiDA: automated interactive infrastructure and database for computational science, Comput Mater Sci 111: 218-30.

8. S Kirklin, JE Saal, B Meredig, A Thompson, JW Doak, et al. (2015) The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies, NPJ Comput Mat 1: 15010-25.

9. T Mayeshiba, H Wu, T Angsten, A Kaczmarowski, Z Song, et al. (2017) The MAterials Simulation Toolkit (MAST) for atomistic modeling of defects and diffusion, Comput Mater Sci 126: 90-102.

10. AH Larsen, JJ Mortensen, J Blomqvist, IE Castelli, R Christensen, et al. (2017) The atomic simulation environment—a Python library for working with atoms, J Phys Conden Matt 29: 273002-32.

11. M Alvarez Moreno, C de Graaf, N Lopez, F Maseras, JM Poblet, et al (2014) Managing the computational chemistry big data problem: the ioChem-BD platform, J Chem Inf Model 55: 95-103.

12. JE Saal, S Kirklin, M Aykol, B Meredig, C Wolverton (2013) Materials design and discovery with high-throughput density functional theory: the open quantum materials database (OQMD), Jom. 65: 1501-09.

13. A Jain, SP Ong, G Hautier, W Chen, WD Richards, et al. (2013) The Materials Project: A materials genome approach to accelerating materials innovation, APL Mat 1: 011002-13.

14. A Jain, G Hautier, CJ Moore, Shyue PO, Christopher C, et al. (2011) A high-throughput infrastructure for density functional theory calculations, Comput. Mater Sci 50: 2295-310.

15. A Jain, SP Ong, W Chen, B Medasani, X Qu, et al. (2015) FireWorks: a dynamic workflow system designed for high-throughput applications, Concurrency Computat.: Pract Exper 27: 5037-59.

16. S Curtarolo, W Setyawan, GLW Hart, M Jahnatek, RV Chepulskii, et al. (2012) AFLOW: an automatic framework for high-throughput materials discovery, Comput Mater Sci 58: 218-26.

17. F Rose, C Toher, E Gossett, C Oses, MB Nardelli, et al. (2017) AFLUX: The LUX materials search API fr the AFLOW data repositories, Comput Mater Sci 137: 362-70.

18. M Scheffler, C Draxl (2014) The Garching Computing Center of the Max-Planck Society, The NoMaD Repository.

19. SP Ong, WD Richards, A Jain, G Hautier, M Kocher, et al. (2013) Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis, Comput Mater Sci 68: 314-319.

20. SP Ong, S Cholia, A Jain, M Brafman, D Gunter, et al. (2015) The Materials Application Programming Interface (API): A simple, flexible and efficient API for materials data based on Representational State Transfer (REST) principles, Comput Mater Sci 97: 209-15.

21. X Yang, Z Wang, X Zhao, J Song, M Zhang, et al. MatCloud: A high-throughput computational infrastructure for integrated management of materials simulation, data and resources, Comput Mater Sci 146: 319-33.

22. CE Calderon, JJ Plata, C Toher, C Oses, O Levy, et al. (2015) The AFLOW standard for high-throughput materials science calculations, Comput Mater Sci 108: 233-38.

23. A Belsky, M Hellenbrandt, VL Karen, P Luksch (2002) New developments in the Inorganic Crystal Structure Database (ICSD): accessibility in support of materials research and design, Acta Crystallogr B 58: 364-69.

24. S Haastrup, M Strange, M Pandey, T Deilmann, PS Schmidt, et al. (2018) The Computational 2D Materials Database: high-throughput modeling and discovery of atomically thin crystals, 2D Mater 5: 042002-37.

25. G Kresse, J Furthm uller (1996) Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, Phys Rev B 54: 11169-86.

26. The Vienna Ab initio Simulation Package: atomic scale materials modelling from first principles.

27. Y Hinuma, G Pizzi, Y Kumagai, F Oba, I Tanaka (2017) Band structure diagram paths based on crystallography, Comput Mater Sci 128: 140-84.

28. W Setyawan, S Curtarolo (2010) High-throughput electronic band structure calculations: Challenges and tools, Comput Mater Sci 49: 299-312.

29. S Plimpton (1995) Fast Parallel Algorithms for Short-Range Molecular Dynamics, J Comp Phys 117: 1-19.

30. LAMMPS Molecular Dynamics Simulator.

31. The open source CFD toolbox, https://www.openfoam.com.

32. D Talia (2013) Workflow systems for science: Concepts and tools, ISRN Software Engineering 2013: 404525-39.

33. K Mathew, JH Montoya, A Faghaninia, S Dwarakanath, M Aykol, et al. (2017) Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows, Comput Mater Sci 139: 140-152.

34. L Ward, A Dunn, A Faghaninia, NE Zimmermann, S Bajaj, et al. (2018) Matminer: An open source toolkit for materials data mining, Comput Mater Sci 152: 60-9.

35. E Gossett, C Toher, C Oses, O Isayev, F Legrain, et al. (2018) AFLOW-ML: A RESTful API for machine-learning predictions of materials properties, Comput Mater Sci 152: 134-45.