

Fast Food for Computational Chemists!

(A programmable cross-platform environment for computational chemist)

V. Ganesh¹

¹Australian National University, Canberra, Australia, ganesh@cs.anu.edu.au

INTRODUCTION

A sizable number of computational chemists (physicists, biologist and material scientists alike) use a wide variety of tools to perform computation, visualization and presentation. Many a times these are incompatible with each other, necessitating reliance on, often cumbersome, external tools to perform appropriate conversions. A few of the computational chemists also use mirid of programming environments to develop new computational codes or visualization tools. However, there is no environment that succinctly amalgamates programming environments and various tools available to a computational chemist, which many a times hinders easy integration and rapid application development. MeTA Studio [1], a cross-platform, programmable environment initially targeted towards computational chemists, intends to address this issue by promoting easy re-usability of codes and ways to develop new applications and methods based on existing code base. The platform consists of components that a computational chemist will need on a day-to-day basis: a molecule visualizer, graph plotting APIs, scripting support, a workspace environment, a collaboration framework and simple networking APIs. MeTA Studio environment can be either used as a tool, or programmed to include additional functionalities that are not already present, allowing researchers to have a fully customizable environment to experiment, collaborate and evolve the base platform.

META STUDIO AS A TOOL

Bread-and-butter tools for supporting day-to-day workflow of a computational chemists are available within MeTA Studio. Figure 1(a) depicts various activities that can be performed in MeTA Studio runtime environment. These activities can be broadly classified as: organize, visualize, code and collaborate. At a deeper level, these activities include: a visualizer, a rendering engine, support for reading common molecular file formats and outputs from popular quantum chemistry programs, a collaboration tool with talk facility to share ideas and data quickly among colleagues and a workspace environment for organizing work related files.

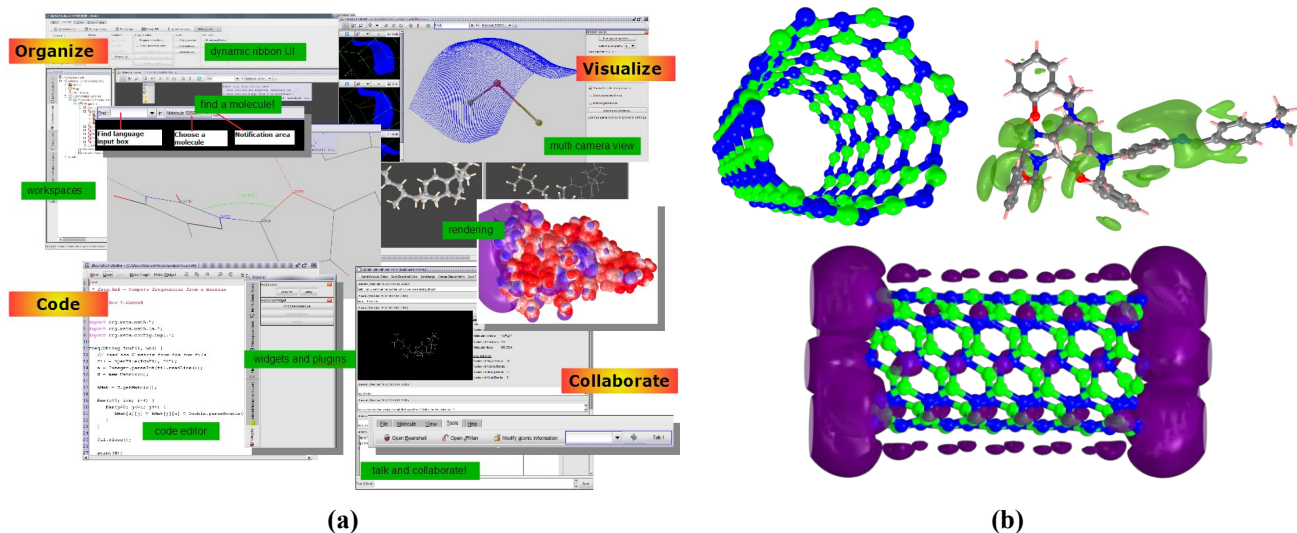


Figure 1: (a) A mash-up of various components in MeTA Studio. (b) Rendered images of molecules and electrostatic potential surface generated form MeTA Studio. See text for explanation.

One of the commonly performed activities of a computational chemist is the visualization (and occasionally rendering) of molecular structures and surfaces. MeTA Studio has extensive support for these activities which either can be used directly or programmed to suite ones needs. Figure 1(b) shows the output from rendered images of a boron nitride nanotube and a cryptand molecule. Apart from standard features offered by most of the molecular visualizers, MeTA Studio visualizer also provides additional tools such as multi-camera view of the molecule and a small “*find*” language that is especially useful while exploring a large molecular structure or a related grid-based property (electrostatic potential or electron density, for instance). A very basic molecule structure editor is also provided that allows one to either sketch small molecular structures or edit an existing molecular structure. The editor also provides access to a basic molecular mechanics based structure optimizer that can be used to obtain a “sensible” 3D structure from a sketch.

META STUDIO PROGRAMMING ENVIRONMENT

Programming in MeTA Studio is supported in a number of ways. The easiest and most intuitive way is provided by a scripting environment which enables one to write plugins or widgets that can be run within MeTA Studio runtime. This scripting environment currently offers programmability via BeanShell, Jython and Scheme. A computational chemist can either use wrapper functions to accomplish a particular task or use the APIs provided to write additional functionalities.

MeTA Studio can also be used as a runtime environment where in user scripts can be executed from command line without invoking the GUI. This is particularly useful to run daemon mode applications (such as performing a series of calculations). A MapReduce [2] implementation based on the networking APIs in MeTA Studio is also provided to enable easy development of distributed applications. An external library framework provided with the IDE makes it relatively easy to use third party libraries such as CDK and JOELib [3].

Lastly, MeTA Studio can also be used as an external library to build completely new applications by reusing components such as the visualizer, or the framework for performing quantum chemical calculation. The ensuing open source environment has wide applicability in various fields including in education. A few of these usage scenarios are enlisted below.

CURRENT USAGE SCENARIOS AND FUTURE SCOPE

MeTA Studio is being currently used in many projects in the form of a tool [4], as a support for a few web applications for computational chemists [5] and as a framework for research into development of quantum chemistry codes for alternative and novel hardwares [6]. MeTA Studio provides a number of additional features to support new kind of divide-and-conquer type algorithms that are turning out to be popular in quantum chemistry space. This is utilized for investigating techniques for fast Hessian computation as well as determining intermolecular hydrogen bond energies [4]. Components such as the molecule visualizer and the widget framework is used in supporting WebProp and WebMTA applications [5]. These applications provide computational chemists an easy access to computing quantum chemical one-electron properties and geometry optimization of molecules. Lately, the quantum chemistry framework in MeTA Studio is being currently used in development and testing of new optimized codes for the Cell architecture [6].

In the retrospect, the “fast food” component in the title of this manuscript is apparently clear at this point, in that, MeTA Studio provides a platform for rapid application development, particularly suited for a computational chemist. A number of components and framework in the IDE (such as networking APIs and related MapReduce) have a broader application area. In future, it is expected that MeTA Studio would be expanded as a generic platform for computational science, going far beyond its current focus on computational structural chemistry.

ACKNOWLEDGMENTS

The author is thankful to Professors Alistair Rendell and Shridhar R. Gadre for their support and useful comments. A number of people have contributed to the MeTA Studio code base in many ways: Anuja Rahalkar, Josh Milthrope and Ritwik Kavathekar to name a few.

REFERENCES

1. “MeTA studio: A cross platform, programmable IDE for computational chemist”, V. Ganesh, *J. Comput. Chem.* **30**, 661 (2009). (Web: <http://code.google.com/p/metastudio/>).
2. “MapReduce: Simplified Data Processing on Large Clusters”, J. Dean and S. Ghemawat, OSDI'04 San Francisco, USA, 2004.
3. (a) CDK, Chemistry Development Kit, A Java library for structural chemo- and bioinformatics, (Web: <http://cdk.sourceforge.net/>). (b) JOELib, A Cheminformatics algorithm library, which was designed for prototyping, data mining, graph mining and algorithm development, (Web: <http://joelib.sourceforge.net/>).
4. (a) “Molecular Tailoring Approach for Exploring Structures, Energetics and Properties of Clusters”, S. R. Gadre, Jovan Jose K. V., and A. P. Rahalkar, *J. Chem. Sci.* (In Press). (b) “Enabling ab initio Hessian and Frequency Calculations of Large Molecules”, A. P. Rahalkar, V. Ganesh, and S. R. Gadre, *J. Chem. Phys.* **129**, 234101 (2009). (c) “Structure, Energetics, and Reactivity of Boric Acid Nanotubes”, M. Elango, V. Subramanian, A. P. Rahalkar, S. R. Gadre, and N. Sathyamurthy, *J. Phys. Chem. A*, **112**, 7700 (2008). (d) “Estimation of N–H···O=C Intramolecular Hydrogen Bond Energy in Polypeptides”, M. M. Deshmukh and S. R. Gadre, *J. Phys. Chem. A*, ASAP, (2009).
5. (a) “WebMTA: A Web-Interface for ab initio Geometry Optimization of Large Molecules using Molecular Tailoring Approach”, R. Kavathekar, S. Khire, V. Ganesh, A. P. Rahalkar, and S. R. Gadre, *J. Comput. Chem.* **30**, 1167 (2009). (b) “WebProp: Web-Interface for ab initio Calculation of Molecular One-Electron Properties”, V. Ganesh, R. Kavathekar, A. P. Rahalkar, and S. R. Gadre, *J. Comput. Chem.* **29**, 488 (2008).
6. “Hartree-Fock code on Cell architecture”, J. Baker, A. P. Rendell and V. Ganesh, under development/ un-published (2009).