Journal of Computational Methods in Sciences and Engineering 12 (2012) 397–406
DOI 10.3233/JCM-2012-0428
IOS Press

IVIChem: An integrative web environment for computational chemistry

Víctor Sojo[a,b], Alexander Peraza[a], Fernando Ruette[a,*], Morella Sánchez[a] and A. Eleonora Acosta[b]
[a]*Laboratorio de Química Computacional, Centro de Química, Instituto Venezolano de Investigaciones Científicas, Caracas, Venezuela*
[b]*Facultad de Ciencias, Universidad Central de Venezuela, Caracas, Venezuela*

Received 8 June 2011
Accepted /Revision 10 October 2011

Abstract. IVIChem is an integrative web environment for computational chemistry that can be accessed globally from any typical computer via the World Wide Web. It assists users all along the research pipeline by facilitating the introduction of molecular geometries, specification of options for a calculation, assembling input files, submission to the calculation queue, monitoring the status of calculations, and graphical analysis of results. IVIChem's graphical user interfaces considerably improve the usability of computational chemistry software in different areas of chemical modeling, and they enhance the performance of researchers by automating the analysis of results and the addition of further calculation packages.

Keywords: IVIChem, user interfaces, GUI, computational chemistry, AgilUs, usability, MOPAC, CATIVIC

1. Introduction

Computational chemistry (CC) can be a complicated discipline, not only due to its advanced theoretical and mathematical background, but also because it uses a great deal of technical vocabulary that only the most experienced of users are fully familiar with, and it requires significant skills for working with both input and output in a variety of poorly interrelated applications. Nevertheless, the number of both dedicated and casual users of CC increases every year, certainly because of the success of the discipline in reaching predictions and explanations within many areas [1,2], including organic chemistry, inorganic chemistry, biochemistry, materials science, electronics, pharmacology, and every other area of chemical research. In particular, results from quantum chemistry (QC) calculations may aid in explaining observed experimental phenomena, as well as in predicting the properties of compounds, as is broadly used in the design and development of new drugs and materials.

A study of a given system of interest may require the use of several CC approaches and corresponding software packages in sub-areas like quantum chemistry (at varying levels of approximation), molecular mechanics, molecular dynamics, chemical kinetics, fluid mechanics, and others. This inherently implies user's choices from potentially thousands of options, thus demanding a good deal of prior knowledge of

*Corresponding author: Fernando Ruette, Laboratorio de Química Computacional, Centro de Química, Instituto Venezolano de Investigaciones Científicas (IVIC), Caracas, Venezuela. E-mail: fruette@ivic.gob.ve.

both the theory and the calculation packages themselves. CC applications generally receive the information for their calculations via text files that must be written in a particular syntax for each package, they are regularly executed via specific commands on the operating system's terminal, and output is generally produced as long text files, such that analyzing results can become an unwieldy task on its own right. Consequently, the usability of CC software is generally poor, making research difficult for both beginners and experts [3–8]. A few CC packages do ship with sophisticated user interfaces, but these normally involve expensive and restrictive licenses and have access limited to only one or a few machines, such that researchers that do not have the possibility to buy them are in a significant disadvantage.

A suitable solution to these usability problems would assist the user in performing every action of the CC pipeline [9]. We here present IVIChem, an integrative web environment of graphical user interfaces chiefly for free CC software that can be accessed worldwide from any typical computer with an internet connection. It assists users at every step of the CC research pipeline, including building molecular geometries, specifying calculation options, running the calculations, monitoring the status of previous calculation jobs and, most importantly, analyzing the results of these calculations. This highly usable interface is easy to learn, and the possibility for administrators to include further CC calculation packages with ease allows not only the improvement of the system, but the establishment of collaborations between institutions.

The difficulties detected in all major steps of the CC research pipeline, namely building geometries, assembling input files, running and monitoring the calculations, retrieving the results, and analyzing the output, were ultimately identified as a general problem of usability of CC software, a solution to which can be expected to significantly accelerate the production of results by researchers while also facilitating the approach of newcomers, particularly students, to the discipline.

It is essential to establish a rational architectural design for the large-scale organization of software for scientific calculations [10], so when developing IVIChem we have considered CC research as a whole process, from the production of input to the analysis of output.

This article describes IVIChem, a free, usable, and expandable integrative web environment for CC that facilitates both scientific research and collaboration. The contribution is organized in the following manner: computational chemistry software, computer languages, and the software development methodology used are presented in Sections 2, 3 and 4, respectively. In Sections 5 to 8 we describe the graphical user interfaces for drawing geometries, dynamically building input files, running and monitoring the status of jobs, and analyzing the results of the calculations. In Section 9 we describe the administrative functions, emphasizing the possibility to incorporate further calculation packages and analysis options. Finally, conclusions and future research directions are presented in Section 10.

2. Computational chemistry software

At present, we have included the semi-empirical calculation packages CATIVIC [11] and MOPAC [12] into IVIChem's user interfaces. Additional applications have been integrated into IVIChem to facilitate both input and analysis, namely JChemPaint [13,14], which has been used to allow directly drawing molecular coordinates in 2D; OpenBabel [15,16] for converting coordinates to the appropriate format for each package; Jmol [17] for visualization of molecular structures in 3D; and Ptplot [18,19] for rendering 2-D plots.

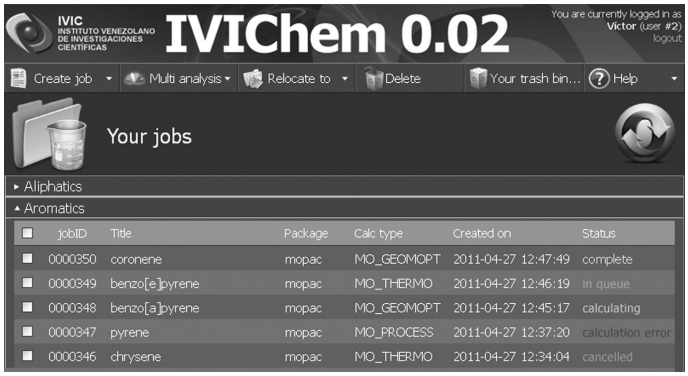


Fig. 1. IVIChem's main screen: a job browser that allows starting new jobs, as well as managing and analyzing previous ones.

3. Computer languages

We have used a variety of computing languages in the development of IVIChem. Being a web application, X-HTML and PHP have been the main of these. PHP has been extensively used on the server side for duties like user validation, processing all the relevant information between pages of the site, interacting with the database, and executing any relevant commands or scripts on either the web server or the calculation clusters. JavaScript is crucial in responding dynamically to user choices on the client's side (the user's computer, as opposed to the web server, where PHP is used); for example, instantaneously converting the web-form choices of the user into valid input files for CC software has been achieved through JavaScript code. CSS (Cascading Style Sheets) has been widely used to control the looks of each of the pages, and to preserve visual styles consistently across the site. MySQL has been used for database management. Python allows processing of output files for easier and more generic analysis. The Unix/Linux Bash language allows file-managing duties like deleting or moving files between different folders and servers, execution of Python scripts, and others.

4. Software development methodology

AgilUs [20], an agile software development methodology [21,22], has been used in the construction of IVIChem. This methodology sets the emphasis of the development on the construction of usability through iterative and incremental development, which implies constant evaluations of user satisfaction with corresponding corrections in the software's functional and non-functional requirements. The construction of usability inherently implies caring for the needs of different users, including experts that understand the details of the calculations but not necessarily the exact algorithms and software that implement them (black-box users), undergraduate students, wet-lab researchers that use CC as a tool for the interpretation of experimental data, CC software development specialists, and researchers in other fields of knowledge related to chemistry. In this sense, IVIChem can be modified to include further calculation options, software packages, properties, and options for automated analysis of results.

5. Drawing geometries

When starting a project, researchers regularly need to procure input geometries from third-party applications, and these geometries must be put down in a syntax that's specific for each package. After logging

```
$CARCAL LEEPAT=0,LEEPMO=0,LEEPAR=0,NUMOLEC=1,$END
$SCF DELENE=1.0D-04,IODAMP=1,$END
$OPT NPASOS=1000,STEP0=0.01,ITIOPT=1,$END
$TITU TITMOL='coronene H CH3',$END
$ARCH $END
$DATMOL NCHARG=1,MULTIP=2,LEEOCU=0,$END
$IMPRI IAVFIN=0,IENECC=0,IMOVIE=1,ICALPRO=0,$END
$LCOR IFRMTO=1,$END
     C          1.2043947  1    3.6588672   1  0.2269837  1
     C          1.2647724  1    2.2434992   1  0.4602151  1
     C         -0.0512294  1    1.4907946   1  0.7949400  1
     C         -1.3270201  1    2.2414343   1  0.3156033  1
     C         -1.2914764  1    3.6651851   1  0.1001700  1
     C          0.0045765  1    4.3307536   1  0.0509515  1
```

Fig. 2. Fragment of an input file for the semi-empirical calculation package CATIVIC. The first eight lines indicate the specifications for the calculation in the particular syntax of this package, whereas the remaining lines contain the atomic coordinates over which the calculation is to be performed (truncated).

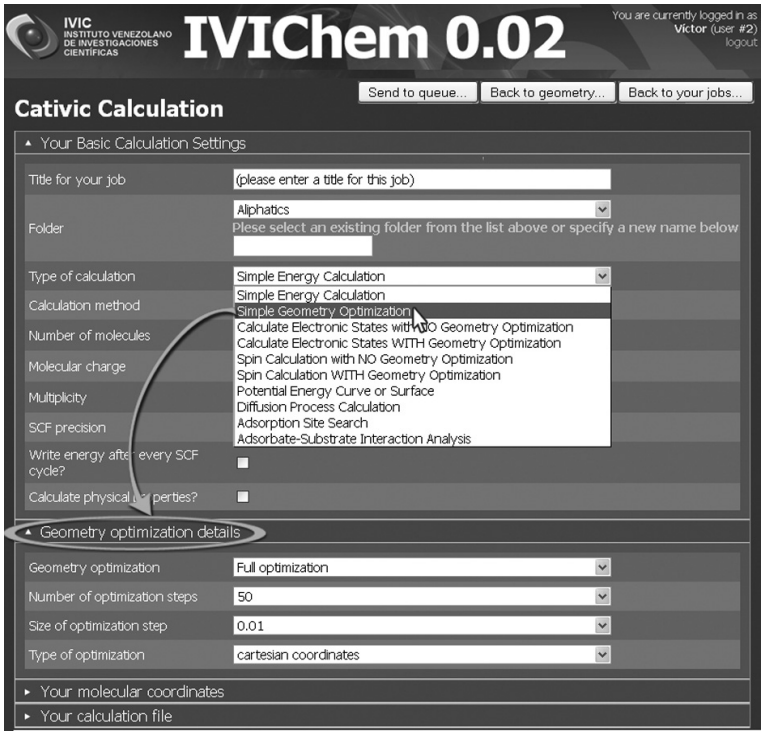


Fig. 3. IVIChem's input builder for CATIVIC readjusting itself upon changing the type of calculation.

in to IVIChem, users first arrive at a job browser screen that shows all previously run jobs and allows both starting new jobs and analyzing previous ones, as shown in Fig. 1. From here, IVIChem users can access a molecular builder that may be used generically for all calculation packages. Although ideally a 3-D molecular builder should be used, a suitable solution is under development so we have temporarily included the 2-D molecular builder JChemPaint [13,14]. IVIChem internally converts the geometries to the proper input format for the chosen package using the powerful libraries of Open Babel [15,16].

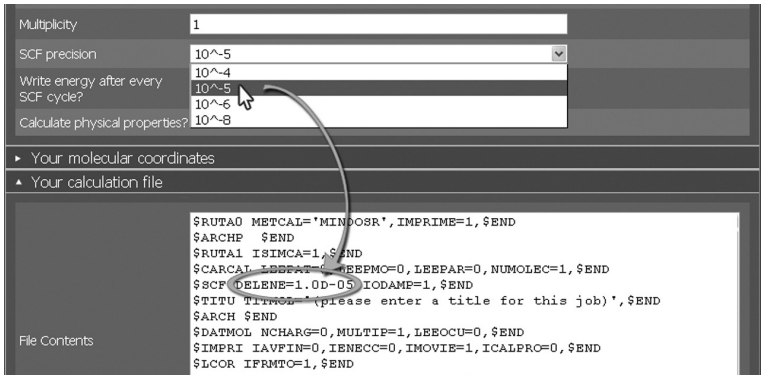


Fig. 4. An immediate effect of a user's selection on the automatic construction of the input file: indicating SCF-cycle precision is not intuitive in CATIVIC so, by providing a simple drop-down list, IVIChem allows the user to focus on the desired specifications for the calculation and not on the proper syntax for the package.

6. Dynamically building input files

CC input is generally varied, complex, multifaceted, and tedious. Once a correct geometry has been procured, a researcher must assemble a calculation file line by line, combining the geometry with the specifications for the calculation, which must in turn be introduced in a typically complicated and unforgiving syntax, exemplified in Fig. 2 for the semi-empirical package CATIVIC [11].

Calculation files for other packages may differ significantly, bearing no resemblance whatsoever to one another. Users are thus forced to always start from a previous file appropriate for the type of calculation to be performed or, worse, learn the syntax of each package by heart. The potential for typographical or syntax errors is large and, indeed, much valuable time is spent learning how to write input files [23].

One of the most important features of IVIChem is the aid provided in building input files. IVIChem users can all but forget the proper syntax for each package, as the interfaces provide dropdown lists, checkboxes, text areas, and other simple input fields that users fill just as they would any typical web form; JavaScript code on the user's side immediately processes the input and dynamically builds the calculation file in response, hiding or showing any relevant fields in response to the choices of the user in the process. In Fig. 3, IVIChem readjusts itself to display the "Geometry optimization details" tab when the user chooses to perform a "Simple Geometry Optimization" calculation. If the user changes back to a "Simple Energy Calculation", the geometry tab is immediately hidden away.

Choices and text input by the user have an immediate effect not only on the fields displayed, but also on the "Your calculation file" field, as indicated by the arrow, as shown in Fig. 4; this last field holds the contents of the input file that will be sent to the dedicated clusters for calculation. Advanced users, however, still have the possibility to alter that final text directly, if desired.

7. Running and monitoring the status of jobs

CC calculation files must normally be sent to a remote dedicated calculation server (i.e., the CC package does not run on the same machine where the input file was assembled), and the progress of the calculation must be monitored somehow.

On IVIChem, once an input file has been dynamically built by assembling the molecular geometries with the specifications from the user's choices on the web form, the job can be submitted to the cal-

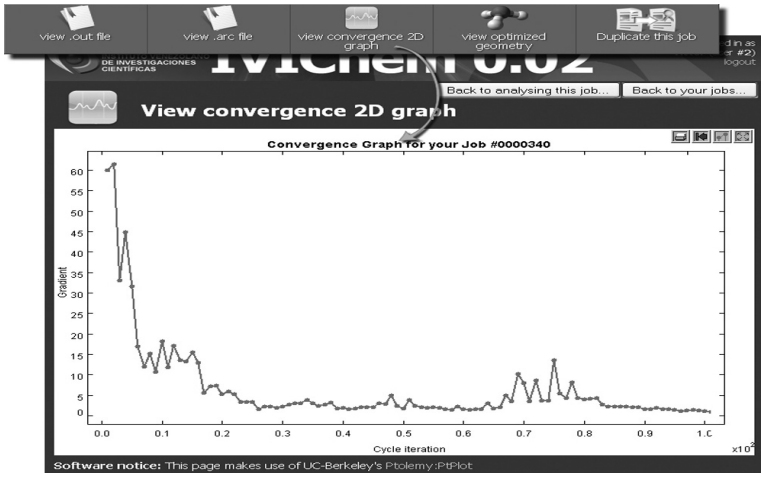


Fig. 5. Options for the analysis of output, and a 2-D plot showing SCF convergence results for a MOPAC calculation. Data is automatically extracted by IVIChem from the output file and displayed on a graph.

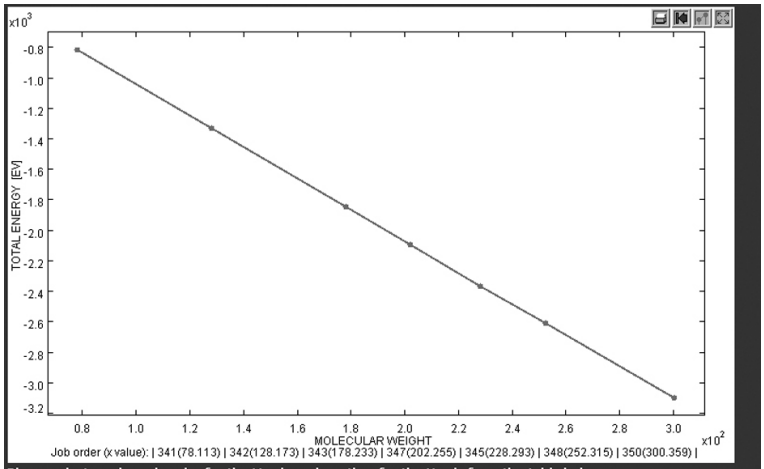
culation queue simply by clicking the button "Send to queue...", which is shown in the top center of Fig. 3.

IVIChem takes care of all file submissions and permissions required, and the status of the job can be easily monitored on the main screen. Five main statuses are possible, namely "complete", "in queue", "calculating", "calculation error", and "cancelled", exemplified in the first five job rows of Fig. 1. A job that is "complete" may be analyzed, a job that is either "in queue" or "calculating" may be "cancelled" by the user, and a job with a "calculation error" status likely contained errors in the input file and so could not be run properly (which normally would only happen if the user modified the input file directly). All five statuses allow a duplication of the input, so both the molecular geometries and the specifications for the calculation can be reused on a new job and these duplicates may be fully modified according to the needs of the user.

8. Analyzing results

In a typical CC research environment, following successful completion of a calculation, the user is left with one or more long text files that need to be retrieved from the remote server and from which all data of interest must then be painstakingly extracted, such that the analysis of output files becomes a demanding task that may take even longer than the calculations themselves. Users must thoroughly search the output file in a text editor, extract the desired information datum by datum, and then tabulate it, plot it, or analyze it in some other way. During the initial scrutiny of our software engineering process, this time-consuming situation was identified as a major cause of frustration for users and, therefore, as a key area for potential improvements in usability.

As depicted in Fig. 1, IVIChem's main screen is a job browser. Clicking on a job links to an analysis page that provides a button bar, shown in the upper part of Fig. 5, for the automated analysis of results. The specific buttons shown depend on the calculation package used, the type of calculation performed, and the status of the calculation (whether or not it has been completed successfully). Upon clicking on an analysis button, IVIChem gathers any necessary files, runs scripts that analyze those files, and displays the desired data visually, disregarding any unwanted information. A 2-D graph, for example,



Job ID	FINAL HEAT OF FORMATION [KCAL]	TOTAL ENERGY [EV]	ELECTRONIC ENERGY [EV]	CORE-CORE REPULSION [EV]	COSMO AREA [SQUARE ANGSTROM]	COSMO VOLUME [CUBIC ANGSTROM]	IONIZATION POTENTIAL [EV]	HOMO ENERGY [EV]	LUMO ENERGY [EV]	NO. OF FILLED LEVELS	MOLECULAR WEIGHT
350	82.90097	-3101.17074	-24884.8635	21783.69277	290.54	339.72	8.320186	-8.32	-1.057	54	300.359
348	78.48778	-2610.9138	-18866.57343	16255.65964	267.35	299.29	8.174476	-8.174	-1.197	46	252.315
347	61.48455	-2094.19018	-13637.31218	11543.122	224.17	243.83	8.346946	-8.347	-1.003	37	202.255
345	80.47133	-2365.60363	-15834.555	13468.95137	259.95	281.37	8.038599	-8.039	-1.320	42	228.293
343	59.49701	-1849.05222	-11087.59349	9238.54127	213.14	223.71	8.36828	-8.368	-0.955	33	178.233
342	40.08241	-1332.43317	-6843.8349	5511.40173	167.15	166.45	8.880796	-8.881	-0.401	24	128.173
341	24.19192	-815.6613	-3282.86403	2467.20272	120.13	108.66	9.637582	-9.638	0.381	15	78.113

Fig. 6. A table displays the molecular energy properties for several MOPAC calculations. The user selects two columns which are then plotted. A clear linear correlation between the total energy and the molar mass can be observed.

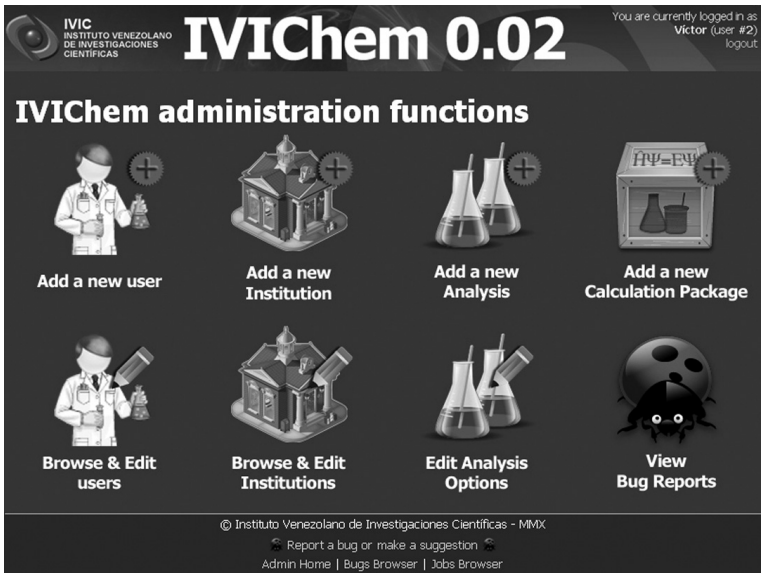


Fig. 7. IVIChem's administrative options allow the enhancement and maintenance of the system.

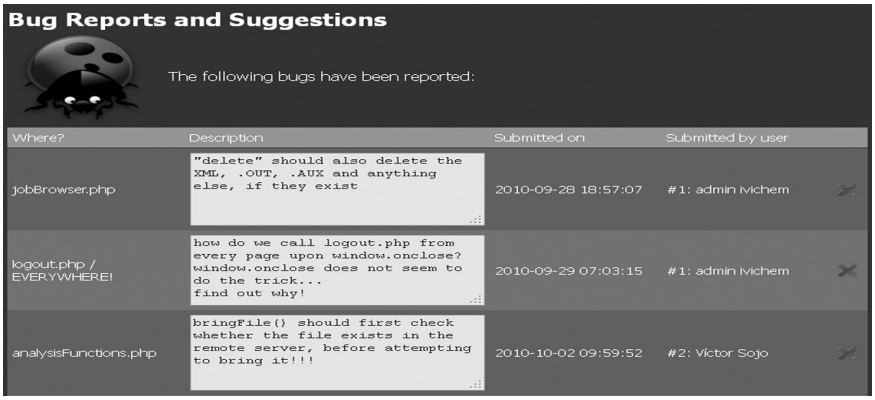


Fig. 8. The bug browser allows the administrator to analyze suggestions and error reports from users, thus accelerating response and consequent improvements to the system.

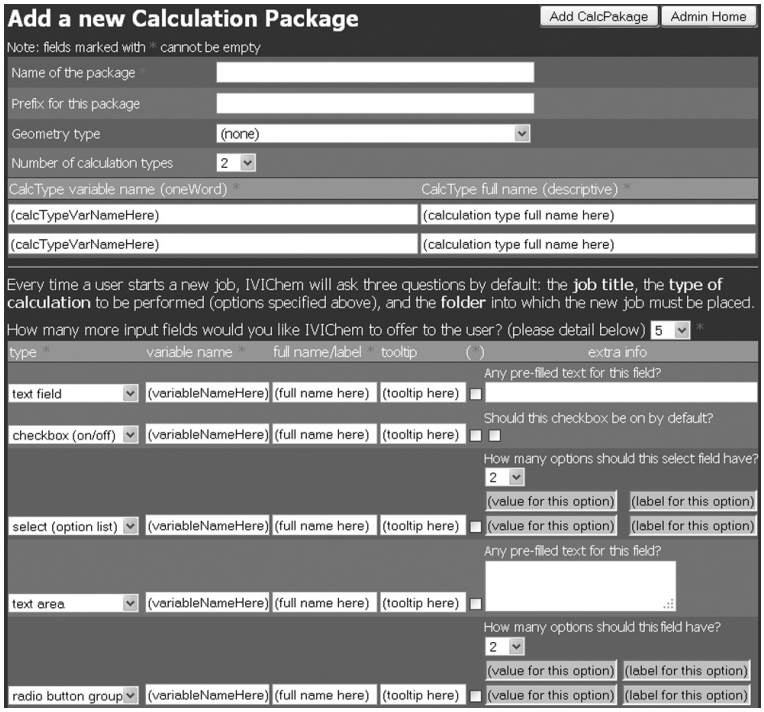


Fig. 9. The form for adding a new package to the system. The administrator chooses the types of input fields to be offered to the user, specifying names and labels for the variables, and tooltips to be shown when the user hovers over the field.

can be plotted using UC Berkeley's PtPlot [18,19] for an energy minimization convergence in MOPAC, as shown in Fig. 5. From the job browser, a user can also select several jobs and perform a simultaneous analysis. As an example, several global energy properties for the calculations are extracted from the output files for each job and then shown in a table (see the lower half of Fig. 6), from which the user can choose two columns to plot and thus compare the results; in the example of Fig. 6, a very clear linear correlation between the total energy and the molecular mass has been found for several polycyclic aromatic hydrocarbons.

9. Administration functions

Typical administration functions include adding, browsing, and editing the information of IVIChem's users and collaborating institutions. These options can be accessed from an analysis main screen, presented in Fig. 7. Administrators can also view bug reports and suggestions (see Fig. 8) submitted by users from an automated interface accessible from every page. Also, additional calculation packages and options for the analysis of results can be added with ease. Regarding the addition of further calculation packages, IVIChem largely automates the process of creating the input-building interfaces by allowing the administrator to specify an arbitrary number of text fields, check boxes (on/off switches), select fields (drop-down option lists), text areas, and radio-button groups, as shown in Fig. 9.

10. Conclusions and future directions

IVIChem is a system of web-based graphical user interfaces that improves the usability of computational chemistry calculation packages (like CATIVIC and MOPAC), facilitating the construction of input files and accelerating the analysis of output. It assists users in the introduction of molecular geometries, construction of input files, running and monitoring calculations, and analyzing results. It also assists administrators in the improvement of the system, by allowing the addition of further calculation packages and analysis options.

Despite this progress, several goals remain to be attained. Efforts are under way to include the *ab initio* packages GAMESS [24] and Firefly (former PC-GAMESS) [25], and we look forward to also including molecular mechanics and dynamics software like GROMACS [26] and Tinker [27]. Options for analyzing catalytic processes at the molecular level, like diffusion, adsorption, desorption, and other reactions that occur on a surface, are in progress. Finally, we are also considering using grid computing and parallelism to allow the globalization of the system.

Acknowledgments

The authors gratefully acknowledge the support of the Venezuelan LOCTI research program.

References

[1] C.J. Manly, S. Louise-May and J.D. Hammer, *Drug Discov Today* **6** (2001), 1101.
[2] I.V. Tetko, J. Gasteiger, R. Todeschini, A. Mauri, D. Livingstone, P. Ertl, V.A. Palyulin, E.V. Radchenko, N.S. Zefirov, A.S. Makarenkoh, V.Y. Tanchuk and V.V. Prokopenko, *J Comput Aided Mol Des* **19** (2005), 453.
[3] K.K. Baldridge, J.P. Greenberg, W. Sudholt, S. Mock, I.Y. Altintas, C. Amoreira, Y. Potier, A. Birnbaum, K. Bhatia and M. Taufer, *Proceedings of the IEEE* **93** (2005), 510.
[4] Y. Podolyan and J. Leszczynski, *Int J Quantum Chem* **109** (2009), 8.
[5] R.J. Gdanitz, G.D. Black, C.S. Lansing, B.J. Palmer and K.L. Schuchardt, *J Comput Chem* **26** (2005), 214.
[6] G. Black, K. Schuchardt, D. Gracio and B. Palmer, *Lectures Notes in Computer Science* **2660** (2003), 122.
[7] A.-R. Allouche, *J Comp Chem* **32** (2011), 174.
[8] Y. Sun, B. Shen, Z. Lu, Z. Jin and X. Chi, *J Comput Aided MolDes* **22** (2008), 119.
[9] W. Sudholt1, I. Altintas and K. Baldridge, *Lectures Notes in Computer Science* **3993** (2006), 69.
[10] F. Losavio, N. Levy, P. Davari and F. Colonna, *Lecture Notes in Computer Science* **3527** (2005), 94.
[11] F. Ruette, M. Sánchez, G. Martorell, C. González, R. Añez, A. Sierraalta, L. Rincón and C. Mendoza, *Int J Quantum Chem* **96** (2004), 321.

[12] J.P. Stewart, *J Computer-Aided Mol Design* **4** (1990), 1.
[13] S. Krause, E. Willighagen and C. Steinbeck, *Molecules* **5** (2000), 93.
[14] C. Steinbeck, Y. Han, S. Kuhn, O. Horlacher, E. Luttmann and E. Willighagen, *J Chem Inf Comp Sci* **43** (2003), 493.
[15] R. Guha, M.T. Howard, G.R. Hutchison, P. Murray-Rust, H. Rzepa, C. Steinbeck, J. Wegner and E.L. Willighagen, *J Chem Inf Model* **46** (2006), 991.
[16] Open Babel. *The Open Source Chemistry Toolbox*. [Online] Septiembre **29** 2010. [Cited: April 25, 2011.] http://openbabel.org/.
[17] Jmol. *Jmol: an open-source Java viewer for chemical structures in 3D*. [Online] August 11, 2010. [Cited: October 17, 2010.] http://jmol.sourceforge.net/.
[18] J. Eker, J.W. Janneck, E.A. Lee, L. Jie, L. Xiaojun, J. Ludvig, S. Neuendorffer, S. Sachs and X. Yuhong, *Proceedings IEEE* **91** (2003), 127.
[19] UC Berkeley EECS Dept. Ptplot, *Ptolemy Project*. [Online] UC Berkeley, 20 Octubre 2010. [Cited: 29 April 2011.] http://ptolemy.eecs.berkeley.edu/java/ptplot/.
[20] *AgilUs: Agile construction of usability (manuscript)*. Acosta, Eleonora.
[21] K. Beck, M. Beedle, A.V. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R.C. Martin, S. Mellor, K. Schwaber, J. Sutherland and D. Thomas, *Manifesto for Agile Software Development*. [Online] 2001. [Cited: April 26, 2011.] http://www.agilemanifesto.org.
[22] K. Beck, M. Beedle, A.V. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R.C. Martin, S. Mellor, K. Schwaber and J. Sutherland, *Principles behind the Agile Manifesto*. [Online] 2001. [Cited: April, 2011]
[23] D.C. Young, Computational chemistry: A practical guide for applying techniques to real-world problems, *John Wiley & Sons* New York, 2001.
[24] M.W. Schmidt, K.K. Baldridge, J.A. Boatz, S.T. Elbert, M.S. Gordon, J.H. Jensen, S. Koseki, N. Matsunaga, K.A. Nguyen, S. Su, T.L. Windus, M. Dupuis and J.A. Montgomery, *Jr J Comp Chem* **14** (1993), 1347.
[25] A.A. Granovsky and K. Kahn, Firefly (former PC GAMESS) Home Page. *Firefly*. [Online] December 4, 2009. [Cited: April 30, 2011.] http://classic.chem.msu.su/gran/gamess/index.html.
[26] E. Lindahl, B. Hess and D.Vander Spoel, *J Mol Model* **7** (2001), 306.
[27] J.W. Ponder, TINKER Home Page. *TINKER Molecular Modeling Package*. [Online] August 24, 2010. [Cited: April 29, 2011.] http://dasher.wustl.edu/tinker/.