The Quixote project: Collaborative and Open Quantum Chemistry data management in the Internet age

Sam Adams[1] , Pablo de Castro[2] , Pablo Echenique[3,4,5] , Jorge Estrada[3,4,6] , Marcus D. Hanwell[7] , Peter Murray-Rust[1] * , Paul Sherwood[8] , Jens Thomas[8] , Joe Townsend[1]

[1]Unilever Centre for Molecular Science Informatics, Department of Chemistry, Lensfield Road, Cambridge CB2 1EW, UK
[2]SONEX Workgroup for Scholarly Output Notification and Exchange, http://sonexworkgroup.blogspot.com
[3]Instituto de Química Física "Rocasolano", CSIC, Serrano 119, E-28006 Madrid, Spain
[4]Instituto de Biocomputación y Física de Sistemas Complejos (BIFI), Universidad de Zaragoza, Mariano Esquillor s/n, Edificio I+D, E-50018 Zaragoza, Spain
[5]Departamento de Física Teórica, Universidad de Zaragoza, Pedro Cerbuna 12, E-50009 Zaragoza, Spain
[6]Departamento de Bioquímica y Biología Molecular y Celular, Universidad de Zaragoza, Pedro Cerbuna 12, E-50009 Zaragoza, Spain
[7]Kitware, Inc., 28 Corporate Drive, Clifton Park, NY 12065, USA
[8]STFC Daresbury Laboratory, Daresbury Science and Innovation Campus, Warrington WA4 4AD, UK

Email: Sam Adams: sea36@cam.ac.uk; Pablo de Castro: pcastro@db.uc3m.es; Pablo Echenique: echenique.p@gmail.com; Jorge Estrada: jorge.estrada@unizar.es; Marcus D. Hanwell: marcus.hanwell@kitware.com; Peter Murray-Rust*:pm286@cam.ac.uk; Paul Sherwood: paul.sherwood@stfc.ac.uk; Joe Townsend: jat45@cam.ac.uk;

*Corresponding author

Abstract

Computational Quantum Chemistry has developed into a powerful, efficient, reliable and increasingly routine tool for exploring the structure and properties of small to medium sized molecules. Many thousands of calculations are performed every day, some offering results which approach experimental accuracy. However, in contrast to other disciplines, such as crystallography, or bioinformatics, where standard formats and well-known, unified databases exist, this QC data is generally destined to remain locally held in files which are not designed to be machine-readable. Only a very small subset of these results will become accessible to the wider community through publication.

In this paper we descrbe how the Quixote Project is developing the infrastructure required to convert output from a number of different molecular quantum chemistry packages to a common semantically rich, machine-readable format and to build respositories of QC results. Such an infrastructure offers benefits at many levels. The standardised representation of the results will facilitate software interoperability, for example making it easier for analysis tools to take data from different QC packages, and will also help with archival and

deposition of results. The repository infrastructure, which is lightweight and built using Open software components, can be implemented at individual researcher, project, organisation or community level, offering the exciting possibility that in future many of these QC results can be made publically available, to be searched and interpreted just as crystallography and bioinformatics results are today.

Although we believe that quantum chemists will appreciate the contribution the Quixote infrastructure can make to the organisation and and exchange of their results, we anticipate that greater rewards will come from enabling their results to be consumed by a wider community. As the respositories grow they will become a valuable source of chemical data for use by other disciplines in both research and education.

The Quixote project is unconventional in that the infrastructure is being implemented in advance of a full definition of the data model which will eventually underpin it. We believe that a working system which offers real value to researchers based on tools and shared, searchable repositories will encourage early participation from a broader community, including both producers and consumers of data. In the early stages, searching and indexing can be performed on the chemical subject of the calculations, and well defined calculation meta-data. The process of defining more specific quantum chemical definitions, adding them to dictionaries and extracting them consistently from the results of the various software packages can then proceed in an incremental manner, adding additional value at each stage.

Not only will these results help to change the data management model in the field of Quantum Chemistry, but the methodology can be applied to other pressing problems related to data in computational and experimental science.

Background

Quantum Chemical calculations and data

High-level quantum chemical (QC) methods have become increasingly available to the broader scientific community through a number of software packages such as Gaussian [1], GAMESS(US) [2], GAMESS-UK [3], NWChem [4], MOLCAS [5] and many more. Additionally, the cost of computer power has experienced an exponential reduction in recent decades and, more importantly, sophisticated approximations have been developed that pursue (and promisingly approach) the holy grail of linear scaling methods [6, 7]. This has enabled any researcher, with no specific QC training, to perform calculations on large, interesting systems using very accurate methods, thus generating a large amount of

valuable and expensive data. Despite the scientific interest of this data and its potential utility to other groups, its lack of homogeneity, organization and accessibility has been recognized as a significant problem by important agents within the scientific community [8, 9].

These problems, and specially the ones related to the accessibility of data have many consequences that reduce the efficiency of the field. As mentioned, QC methods are computationally expensive: the scaling of the computer effort and storage of high-level computations with the size of the system (N) is harsh, reaching, for example, N^7, for the most expensive and most accurate wavefunction-based methods, such as Coupled Cluster [10–12]. This makes it very difficult for groups that cannot use supercomputing facilities to have access to high-quality results, even if they possess the expertise to analyze and use the data. Even groups that do have access to powerful computational resources, given the lack of access to previously computed data by other researchers, often face the choice between *two inefficient* options: either they spend a lot of human time digging in the literature and contacting colleagues to find out what has already been calculated, or they spend a lot of computer effort (and also human time) calculating the needed data themselves, with the risk of needlessly duplicating work.

Another problem originating in the lack of access to computed QC data and the very large number of methods available, is that users typically do not have the integrated information about which method presents the best accuracy *vs.* cost relation for a given application. The reason is that comparing one quantum chemical method with another, with classical force fields or with experimental data is non-trivial, the answer frequently depending on the studied molecular system and on the physical observable sought. Moreover, all the details and parameters that define what John Pople termed a *model chemistry* [13], *i.e.*, the exact set of rules needed to perform a given calculation do not obey a continuous monotonic function. Thus increasing the expense and "accuracy" of a calculation may not always converge to the "correct" solution. As a consequence, the quality of the results does not steadily grow with the computational effort invested, but rather there exist certain tradeoffs that render the relation between them more involved [14–16]. Hence, not only the choice of the more efficient QC method for a given problem among the already existing ones, but also the design of novel model chemistries becomes 'more an art than a science' [17], based more on know-how and empiricism than in a set of systematic procedures.

Design of Scientific data repositories

In this paper we describe a novel, flexible, multipurpose repository technology. It arises out of a series of meetings and projects in the computational chemistry (compchem) community which have addressed the

desire and need to have repositories available for capturing and disseminating the results of QC calculations. It is also strongly influenced by the eScience ("cyberinfrastructure", "eResearch") programs which have streesed the value of instant semantic access to research information from many disciplines, and by the Open Innovation vision supported by the Scientific Software Working Group of CECAM (Centre Européen de Calcul Atomique et Moléculaire)[1], which seeks an innovation model based on sharing, trust and collaboration, and which recognizes the important role played by the availability of reference data and archives of outputs of calculations and simulations. It also coincides with the increasing mandates for data publication from a wide range of funders; our repository can address a large part of these requirements. This paper describes a distributed repository technology and the social aspects associated with developing its use. The technology is robust and deployed but the way it may be used is at a very early stage. We address known social issues (sustainability, quality, etc.) but expect that deployment, even in the short term, may look very different from what is reported.

The development and acceptance of Wikipedia may act as a valuable guide and it represents a community-driven activity with community-controlled quality. Although variable, we believe that articles for most mainstream physical sciences are reliable. Thus to help understand and represent moments of inertia in computational chemistry we can link to Wikipedia (http://en.wikipedia.org/wiki/Moment_of_inertia). This contains many hundreds of edits over eight years from many authors - it is almost certainly "correct". Quixote has many of the same features - anyone can contribute content and repurpose it. We expect a culture to emerge where the community sets guidelines for contributions and corrections/annotations. We are building filters ("lenses") so that the community can identify subcollections of specific quality or value.

The background to Quixote includes a number of meetings and projects which specifically addressed the development of infrastructure in computational chemistry and materials. The goal of these was to explore the commonality between approaches and see how data and processes could interoperate. One (Materials Grid) also addressed the design and implementation of a repository for results.

- 2004: A meeting under the UK eScience program "Toward a common data and command representation for quantum chemistry" (http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=394).
- 2006: A meeting under the auspices of CECAM "Data representation and code interoperability for

[1] http://www.cecam.org/

computational materials physics and chemistry" (http://www.cecam.org/workshop-50.html)

- 2005-2010: A 5-year project under the COST D37 program to develop various aspects of interoperability both within the calculation (Q5COST) and between programs (WG5).
- A funded project in computational materials ("Materials Grid") (http://www.materialsgrid.org/) which resulted in considerable development of CML specifications and trial implementations in a number of codes (CASTEP, DLPOLY).

These meetings and projects were exploratory and localized. Within them there was a general agreement that interoperability and access to results would be a great benefit. But they also highlighted the problem that infrastructure development is expensive and, if public, requires political justification for funding. Such funding is perhaps most likely to come from supranational efforts such as computational Grids, where there is a clear imperative for making services as accessible as possible. In COST-D37 the funding was for meetings and interchange visits; the WG5 community made useful but limited progress without dedicated developer or scientist funding.

There is often a vicious circle here - a frequent reason for not adopting a new technology in chemistry is "there is no demand for it". This becomes a self-fulfilling prophecy and naturally limits innovation. It is also true that people are often only convinced by seeing a "working system" - hypothetical linkages and implementations have often been wildly optimistic. Therefore without seeing a working repository it is difficult to know what its value is, or the costs of sustaining it.

However the Internet age shows that it is much easier, cheaper and quicker to get new applications off the ground. It should be possible, in a short time and with modest effort, to create a system which demonstrates semantic interoperability and to convince a community of its value. We have successful examples of this reported elsewhere in this issue (OSCAR, CrystalEye, Open Bibliography) where an early system has caught the imagination and approval of a section of the community.

The general need for data repositories

These issues, and undoubtedly more that will appear in the future, together with a wealth of scientific problems in neighbouring fields, could be tackled by public, comprehensive, up-to-date, organized, on-line repositories of computational QC data. Additionally, several fields reporting experimental data require it to be presented in a standard validatable form. The crystallography community has long required deposition of data as a prerequisite for publication, and this is now enhanced by machine validation (the

CheckCIF philosophy and program[2]). When data are submitted, the system can comment on whether all appropriate data are present, inspect their values and compare either with known ranges or re-compute relationships between them based on accepted theoretical principles. In this way reviewers and readers can expect that a very large number of potential errors in experiment and publication have been eliminated. This requirement for deposition of data as part of the publication process is increasingly common in bioscience, like genetics or proteomics, where the NCBI GenBank[3] or the Protein Data Bank (PDB) [4] constitute very successful examples of data sharing and organization. In an age in which both the monetary cost and the accuracy of QC calculations rival those of experimental studies, the need to extrapolate the model to this field seems obvious. We also note that funders are requiring that data be deposited as part of the condition of funding.

On the one hand, there exist some in-house solutions that individual research groups or firms have built in order to implement a local-scale data management solution. This is the case of David Feller's Computational Results Database[5] [18], an intra-lab database to store and organize more than 100,000 calculations on small to medium-sized molecules, with an emphasis on very high levels of the theory. Also, the commercial standalone application SEURAT[6] can open and parse QC data files and allows for metadata customization by the user, thus providing some limited, local databasing capabilities. In the same family of solutions, ChemDataBase [19] is a data management infrastructure mainly focused on virtual screening which presents the distinctive feature of being able to create and retrieve databases over grid infrastructures. Packages for interacting with QC codes (launching, retrieving and analyzing calculations), such as ECCE[7] or Ampac[8], have modest data management capabilities too, although only insofar as it helps to perform their main tasks, and they can be regarded as intra-lab solutions as well. Probably the most complete in-house infrastructure of which we are aware of is the RC^3 (Regional Computational Chemistry Collaboratory) developed by the group of David Dixon at the Department of Chemistry of the University of Alabama. The main objective of RC^3 is to perform the everyday data backup, collection and metadata assignment for calculations, and to organize them for research purposes. At the time of writing, RC^3 has been tested by 36 users for more than a year, and backed-up and organized 1.6 million files, amounting to 1.5TB of data storage. The database contains 144,000 records and it can

[2] http://checkcif.iucr.org/
[3] http://www.ncbi.nlm.nih.gov/genbank/
[4] http://www.rcsb.org/pdb/home/home.do
[5] http://tyr3.chem.wsu.edu/~feller/Site/Database.html
[6] http://www.synapticscience.com/seurat/
[7] http://ecce.emsl.pnl.gov/index.shtml
[8] http://www.semichem.com/ampac/afeatures.php

currently parse multiple QC data formats.

Heterogeneous data repositories

A different category of data management solutions from the one discussed above is that constituted by a number of online web-based repositories of QC calculations, normally developed by one research group with a very specific scientific objective in mind. Among them, we can mention the NIST Computational Chemistry Comparison and Benchmark DataBase (CCCDB)[9], which contains a collection of experimental and calculated *ab initio* thermochemical, vibrational, geometric and electrostatic data for a set of gas-phase atoms and small molecules; the Benchmark Energy and Geometry DataBase (BEGDB)[10] [20], which includes geometry and energy CCSD(T)/CBS calculations as well as other high-level calculations, with a special emphasis on intermolecular interactions; the DFT Database for RNA Catalysis (QCRNA)[11] [21], which contains high-level density-functional electronic structure calculations of molecules, complexes and reactions relevant to RNA catalysis; the Atomic Reference Data for Electronic Structure Calculations[12] [22] compiled at NIST, containing total energies and orbital eigenvalues for the atoms hydrogen through uranium, as computed in several standard variants of density-functional theory; or the thermochemistry database at the Computational Modeling Group of Cambridge's Department of Chemical Engineering[13], collecting thermochemical data of small molecules, powered by RDF and SPARQL and offering the output files of the calculations, together with the parsed CML[14] [23].

Apart from these solutions (either local or web-based), in which one or a few groups build a complete data management infrastructure, one can also consider the possibility of adopting a modular approach, in which different researchers tackle different parts of the problem, whilst always enforcing the maximum possible interoperability between the modules. The Blue Obelisk group[15] [24] has been championing this approach for a number of years now, and many of the developers of the tools discussed below are members of it. In this category of solutions, we can also mention the Basis Set Exchange (BSE)[16] [18, 25], which provides an exhaustive list and definition of the most common basis sets used in QC calculations, thus facilitating the definition and implementation of semantic content regarding the method used, as well as improving the interoperability among codes at the level of the input data; modern tagging and markup technologies like

[9] http://cccbdb.nist.gov/
[10] http://www.begdb.com/
[11] http://theory.rutgers.edu/QCRNA/
[12] http://www.nist.gov/pml/data/dftdata/index.cfm
[13] http://como.cheng.cam.ac.uk/index.php?Page=cmcc
[14] http://cml.sourceforge.net
[15] http://www.blueobelisk.org/
[16] https://bse.pnl.gov/bse/portal

XML and RDF together with the building of semantic dictionaries, not only to promote interoperability, but to do it in a web-friendly manner that allows one to easily plug modules and build complex online data management projects; the CML language (a chemical extension of XML) [23] is also one of the few cases in which a common semantics has been widely adopted by the chemistry community, and its extension to the QC field is one of the cornerstones of the Quixote project described here. Also on the interoperability front, we can mention the cclib[17] [26] and CDK[18] [27] libraries, as well as the OpenBabel toolbox[19], which provide many capabilities for reading, converting and displaying QC data in many formats. Regarding the ease of use of possible data management solutions, the Open Source molecular editor and visualizer Avogadro[20] can certainly be used as a useful module in complex projects, and in fact the design of Quixote is being carried out in collaboration with the developers of Avogadro, with the intention of efficiently interfacing it in future versions. The Java-based viewer Jmol[21] performs similar tasks.

All in all, and despite the numerous efforts described above, it is clear that a global, unified, powerful solution to the management of data in QC does not exist at present; at the same time that the new internet-based technologies, the existence of vibrant communities, and the wide availability of powerful software to perform the calculations, and to convert and analyze the results, all seem to indicate that the field is ripe to produce a revolutionary (and much needed) change in the model. In this article, we present the beginnings of an attempt to do so.

The Quixote solution

The catalyst for Quixote was a meeting on interoperability and repositories in QC held at ZCAM (Zaragoza Scientific Center for Advanced Modeling), Zaragoza (Spain) in September 2010. There was general agreement on the need for collection and re-dissemination of data. In the final discussion a number of participants felt that there was now enough impetus and technology that something could and should be done. This wasn't a universal view, and we are aware that Quixote is unconventional in its genesis and aspirations – hence the name, reflecting a difficult but hopefully not impossible dream.

We decide to pursue this as an informal "unsponsored" project. It is not actually "unfunded", in that we recognize the critical and valuable cash and in-kind support of several bodies, including CECAM, STFC Daresbury Laboratory, EPSRC, JISC, ZCAM, and the employers of many of the participants. In particular

[17] http://cclib.sf.net
[18] http://cdk.sf.net
[19] http://openbabel.org
[20] http://avogadro.openmolecules.net
[21] http://jmol.sourceforge.net/

we have been able to hold, and continue to hold, meetings. But there are no sponsor-led targets or requirements . In this it has many of the features of successful virtual projects in ICT (such as Apache, Linux, *etc.*) and communal activities such as Wikipedia and Open Street Map.
Speed and ambition were critical and project management has been by deadlines – external events fixed in time for which the project had to have something to show. These have included:

- An *ad hoc* meeting in 2010-10 in Cambridge where a number of the participants happened to be. This was to convince ourselves that the project was feasible in our eyes
- The PMR symposium 2011-01 that has catalysed this set of articles
- A workshop 2011-03 at STFC Daresbury Laboratory to demonstrate the prototype to a representative set of QC scientists and code developers
- Open repositories (OR11) 2011-06 where the technology was presented to the academic repository community as an argument for the need for domain repositories
- (planned) A meeting in Zaragoza 2011-08 where the argument for domain repositories will be demonstrated by Quixote.

As of 2011-06 we have a working repository with over 6000 entries, which are searchable chemically, by numeric properties and through metadata.
Our primary goal has been to build working, flexible technology without being driven by specific use-cases. This can be seen as heresy, and indeed we might regard it as such ourselves, if it were not that we have spent about 10 years working in semantic chemistry, computational chemistry and repositories and so have anticipated many of the possible use cases and caveats. To help show Quixote's flexibility we now list a number of use cases, any one of which may serve to convince the reader that Quixote has something to offer:
The Quixote system (Figure1 shows the workflow, Figure 2 shows the distributed heterogeneity) is very flexible in that it can be installed in several different ways. Here we give a number of possible uses of the system, some of which we have deployed and several more we expect to be useful.

- Collection of results within a group or laboratory. There is a growing desire to capture scientific results at the time of creation, and we have been involved in several projects (CLaRION, JISC XYZ) the impetus of which is to see whether scientists can capture their data as they create it.

Computational chemistry is one of the simplest types of results and Quixote has been designed so that a single log file provides most of the input to the repository. This system allows groups and individual researchers to "pick up their results" and transport them to different environments.

- Formal publication in journals and theses. Results in a Quixote repository can be made available to other people and parties in the publication process. For example an author could make their results available to a journal before review so that the editors and reviewers could use the data to assess the value of the science. Similarly a graduate student could make their results available as part of their thesis submission and these could be assessed by the examiners. If the thesis and accompanying data are also published in the institutional repository then this provides a simple but very effective way of capturing and preserving the record of scientific experiments.

- Teaching and learning resources. Quixote can collect resources used for teaching and can also be used to provide subsets of research objects which are valuable for teaching and learning. For example in the current set there are 75 calculations on benzene, mainly from Henry Rzepa's laboratory and these have been deposited by students carrying these out as part of their undergraduate work. This resource allows us to compare methods and to get information and experience which may help us do similar calculations.

- A collaborative central repository for a project. An increasing number of projects are distributed over geography and discipline. (The current Quixote project is an example.) A repository allows different people and groups in the project to share a central resource in an analogous manner to the use of Bitbucket and similar repositories for sharing code.

- A set of reference data and molecules. Quixote allows us to search for different parameters used in a given problem (*e.g.* level of theory, number of orbitals, convergence of results, algorithms, *etc.*).

- Validation sets for software and methods. In a similar manner datasets within Quixote can be used by different groups as reference input to compare results from different programs or different approaches.

- Enrichment of data through curation. Quixote is annotatable, so that it is possible for the world community to add their comments to particular entries. If a result is suspect, an annotation can be added. Similarly it is possible to point out related entries highlighting different scientific aspects.

- Building blocks for calculations. It is often valuable to start from an unknown program resource (*e.g.*

a molecule whose structure is known and where the calculations are verified) and to modify it slightly for a related calculation, *e.g.* by adding additional atoms or by refining the calculation parameters.

- Combining data from different sources. As Quixote can also store experimental structures such as crystallographic ones, or experimental data such as spectra it is possible to enhance and combine components of the calculation.

- Data-driven science. Now that computational chemistry is relatively cheap and relatively accessible for a very large number of scientists, we foresee that literally millions of processors will be used routinely to calculate theoretical chemistry results. This allows us to carry out data mining from the Quixote repositories with the possibility of discovering new scientific patterns.

- Indexing the web. In a similar way to our indexing of crystallography through CrystalEye[22] we anticipate that web crawlers can increasingly discover and retrieve published computational chemistry.

- Developing software tools. Since Quixote represents an abstraction of many codes, developers writing software for computational chemistry will be able to see the type of semantics which are captured and the structure of the document.

Quality

The collection of the scientific computional record through Quixote could be regarded as an objective process in that each logfile is sufficiently described from the view of repeatability. Any user of Quixote could, if they had access to the code(s), re-run the calculation and "get the same output". The examples of student calculations on benzene in the current content illustrate this view.

On the other hand it can be objected that unless a calculation is carried out with professional care then it can not only be meaningless but seriously misleading. Non-experts in QC can obtain these results and can misinterpret them. This is true, but it is a fact of modern Open science – results should be and are available to anyone. Science must evolve social and technical methods to guide people to find the data they want. We can buy a kit and in our garages determine the sequence of a gene or protein without realising the potential experimental errors, or the difficulty of describing the species or strain that it came from. We can buy table-top crystallography sets that will automatically solve the structure of almost all crystalline materials. The results of these experiments are valuable if interpreted correctly and much of the time there

[22] http://wwmm.ch.cam.ac.uk/crystaleye/

is little room for serious error. However we might not realise that one lanthanide might be mistaken for another, that crystals can be twinned, and that certain spacegroups are problematic. Similarly the neophyte may not appreciate the difficulty of getting accurate energies, spin densities, non-bonded interactions, and many more subtleties of computational chemistry. But Pandora's box has been opened and computational chemistry is a commodity open to all. Quixote will help us in making our communal judgments.

There are a few objective concerns about quality. The Quixote system converts legacy computational chemistry (logfiles) into semantic form. Automatic conversion will usually have a small number of errors, but mainly in that fields will not be recognized, rather than corrupted. In the early stages the semantics of some quantities may be misinterpreted (many are often laconic "E=1.2345" - what exactly is E? and what are the units?) Given the exposure of the system to "many eyes" such problems will be few and should be relatively rapid to remove.

The fuzzier concern is whether Quixote can grow to gain the confidence of the QC and the non-QC community. Computational chemistry has the unique feature that anyone in the world, given the same input, will create the same output. The question is not whether the log file is an accurate record of the calculation but whether the calculation is valuable. It is quite possible to create junk, often unknowingly, and the commonest way is by inputting junk. A typical example is that many chemoinformatics programs can garble hydrogen counts and formal charges. However there are several criteria that the Quixote user and community can apply:

- If the methodology is very standard, then the results are likely to be usable in a similar way to other results using the same method. For example a very common combination of method and basis for organic molecules is B3LYP + 6-31G**. If another group has successfully employed this for a set of molecules similar to the user's it is likely to be a useful starting point. This does not of course absolve the user from critical judgement but it is better than having nowhere to start.

- Automated methods can be used to compare the results of calculations for similar molecules or with varied parameters.

- We particularly encourage collections provided by specified individuals or groups. We have made two available in the current release (Dr. Anna Croft, Prof. Henry Rzepa). The user can browse through collections and get an idea of the type of calculation and the quality of metadata.

- Are the data coupled to publication? In CrystalEye almost all records are coupled to primary publications which can be read by the user (assuming that they have access to the journal). There is no technical barrier why this should not be done for articles and theses in computational chemistry. This is harder in compchem until the community develops a culture of publishing data concurrently with articles.

- Have the entries been annotated? This feature will shortly be available in Quixote, probably through blogging tools.

- Are there criteria for depositing an entry in the particular Quixote repository? Since we expect there to be many repositories, some of them can develop quality criteria for deposition. Some, perhaps the majority, may have human curators. In the first instance it will be important that users can assess the quality of a particular Quixote repository and we are appealing to any scientist who have collections of computational chemistry data that they would be prepared to make available. We expect that there will be a range of levels of quality in Quixote repositories. For example a crawler visiting random web sites for data might store these in an "unvalidated" repository. Users could examine this for new interesting entries and make their own decisions as to their value. The web has many evolved systems for the creation of quality metrics (popularity, usage, recommendations, *etc.*) and many of these would make sense for compchem. A journal might set up their own repository (as is done for crystallography). A department could expose its outputs (and thereby gain metrics and esteem) and the contents would be judged on the assessment of the creators.

Methods

All materials and methods mentioned here are available as Open Source/Data from the Quixote site or the WWMM Bitbucket repository. A small amount is added as appendixes to guide the reader.

Concepts and vocabulary

In any communal system requiring interoperability and heterogeneous contributions it is critical to agree concepts and construct the appropriate infrastructure. Chemistry has few formal shared ontologies and Quixote explores the scope and implementation of this for QC.

We draw inspiration from formal systems such as the Crystallographic Information File (CIF) created over many years by the International Union of Crystallography (IUCr). This is a community activity with

medium-strong central management - the community has an input but there are formal procedures. It works extremely well and is universally adopted by crystallographers, instrument manufacturers, and publishers. The vocabulary and semantics have been developed over 20 years, are robust and capable of incremental extension. We take this as a very strong exemplar for Quixote and more widely QC.

We believe that almost all QC codes carry out calculations and create outpus which are isomorphic with other codes in the community. Thus an "electric dipole", "heat of formation" or a "wavefunction" is basically the same abstract concept across the field. The values and the representation will be code-dependent but with the appropriate conversions of (say) units, coordinate systems and labelling, it is possible to compare the output of one code with another. This is a primary goal of Quixote, and we work by analysing the inputs and outputs of programs as well as top-down abstractions. It also means that Quixote is primarily concerned with what goes into and comes out of a calculation rather than what is held inside the machine (the data model and the algorithms).

Community development

From the human resource point of view, the Quixote project operates on a decentralised approach with no central site and with all participants contributing when available, and in whatever quantity they can donate at a particular time. For that reason, different parts of the project progress at variable speeds and technically independently. This means that there is very little effort required in collating and synthesising other than the general ontological problem of agreeing within a community the meaning deployment and use of terms and concepts.

The work is currently driven (*cf.* use cases) by datasets which are available. This drives the need to write parsers, collate labels into dictionaries, and collate results. In the week of 2011-05-09, for example, we ran daily Skype conferences, with Openly editable Etherpads[23] generously provided by the Open Knowledge Foundation (OKF)[24]. The participants created tutorial material, wiki pages, examples and discussions which over the week focused us to a core set of between 20-50 dictionary entries that should relate to any computational chemistry output. The input to this effort was informed by logfiles from the Gaussian, NWChem, Jaguar and GAMESS-UK programs.

The initial approach has been to parse logfiles with JUMBO-Parser, as this can be applied to any legacy logfiles and does not require alterations of code. (At a later date we shall promote the use of CML-output

[23] http://quixote.wikispot.org/
[24] http://okfn.org/

libraries in major codes.) At this stage it is probably the best approach to analyse the concepts and their structure. A JUMBO-Parser is written for each code and run over a series of example logfiles. Ideally every part of every line is analysed and the semantic content extracted. In practice each new logfile instance can bring novel structure and syntax but it is straightforward to determine which sections have been parsed and which have not. Parsing failure may be because a parser has not been written for those sections, or because the syntax varies between different problems and runs. The parser writer can then determine whether the un-parsed sections are important enough to devote effort to, or whether they are of minor importance and can be effectively deleted.

The process is highly iterative. The parser templates do not cover all possible document sections and initially some parts remain unparsed. The parsers are then amended and re-run; it is relatively simple in XML to determine which parts still need work.

Currently (2011-06) there are about 200 templates for NWChem, 150 for Gaussian and a small number for Jaguar, GAMESS-UK, GAMESS(US), AMBER and MOPAC[25]. Each time a parse fails, the section is added as a failing unit test to the template and these also act as tutorial material and a primary source of semantics for the dictionary entries.

Quixote components

JUMBO-Converters

The JUMBO-Converters are based on a templating approach, matching the observed output to an abstraction of the QC concepts. They have been hand-crafted for a number of well-structured output files (Gaussian archive files, MOPAC and various punchfiles) but the emphasis is now on writing JUMBO-Parsers for the logfiles for each code. We have explored a wide range of technologies for parsing logfiles including machine learning, formal grammars (lex/yacc), ANTLR[26], but all of these have problems when confronted with unexpected output, variations between implementations, error messages and many other irregularities. The JUMBO-Parser will not be described in detail here but in essence consists of the following approach:

- Recognition of common document *fragments* in the logfile (*e.g.*, tables of coords, eigenvalues, atomic charges, *etc.*) which appear to be produced by record-oriented (FORTRAN format) routines in the source code. We create a *template* for each such *chunk*, which contains *records*, with regexes for each

[25] http://openmopac.net
[26] http://www.antlr.org/

record that we wish to match and from which we will extract information. These templates can be nested, often representing the internal structure of the program (*e.g.*, nested subroutine calls).

- Each template is then used to match any chunks in the document, which are then regarded as completed and unavailable to other templates. The strategy allows for nesting and a small amount of back-tracking.

- Chunks of document that are not parsed may then be extracted by writing additional parsers, very often to clean up records such as error messages or timing information.

At the end of this process a good parse will results in a highly-structured document with CML module providing the structure and CML scalar, array and matrix providing the individual fields[27].
This document is rarely fit for purpose in Quixote or other CML conventions and a second phase of transformation is applied. This carries out the following:

- Removal of unwanted fields.

- Removal of unnecessary hierarchy (often an artifact of the parsing strategy)

- Addition of `dictRef`s to existing dictionaries

- Addition of units (often not explicitly mentioned in the logfile but known to the parser writer)

- Grouping of sibling elements into a more tractable structure (unflattening)

- Annotation of modules to reflect semantic purpose, *e.g.*, initial coordinates, optimizations, *etc.*

- Re-structuring of the modules in the parsed output to fit the *compchem* convention[28]

This is carried out by a domain-specific declarative language which makes heavy use of XPath and a core set of Java routines for generic operations (delete/create/move elements, transform (matrix/molecule/strings etc.)). This approach means that failures are relatively silent (a strange document does not crash the process) and that changes can be made external to the software (by modifying the transformation files). As with the templates this should make it easier for the community to maintain the process (*e.g. when new syntax or vocabulary occurs).*
A typical template is shown in Appendix 2.

[27] http://quixote.wikispot.org/Tutorials_and_problems
[28] http://www.xml-cml.org/convention/compchem

CML Conventions and Dictionaries

The final output is CML compliant to the compchem *convention and validated against the current validator*[29]*. The dictionaries are in a constant state of update and consist of a reference implementation on the CML site and a working dictionary associated with the JUMBO-Converters distribution. As concepts are made firm in the latter, they are transferred to the reference dictionary.*
The current compchem dictionary is shown in Appendix 1. It contains about 90 terms which are independent of the codes. We expect that about the same amount again will be added to deal with other properties and solid state concepts.

Lensfield2

Lensfield2[30] *is a tool for managing file transformation workflows and can be thought of as a* `make` *for data. Lensfield2 requires a build file, defining the various sets of input files and the conversions to be applied to them. Like* `make`*, for instance, Lensfield2 is able to detect when files have changed, and update the products of conversions depending on them. However, unlike* `make` *where this is just done through comparison of files' last-modified times, Lensfield2 records the complete build-state, so is able to detect if intermediate any change in configuration, such as when the parameterisation of builds has changed, and when versions of tools involved in the various steps of the workflow are updated or if intermediate files are altered.*
Lensfield2 is designed to run workflow steps written in Java and build using Apache Maven[31]*, utilising Maven's dependency management system to pull in the required libraries for each build step.*
*Lensfield2 has been successfully used in running the parser and subsequent software over the 40,000 files in the test datasets 1-4 (*v.i.*).*

RESTful uploading

It is important that the methods for "uploading" and "downloading" files are as flexible as possible. Some collaborators may not have privilleges to run their own server, so they need to be able to upload material to a resource run by other collaborators. However, if the protocols are complex then they may be put off taking part. Similarly, others may wish to delegate this to software agents which poll resources and aggregate material for uploading. Similar variability exists in the download process. Web-based collaborators are becoming used to very lightweight solutions such as Dropbox[32] *where files can be uploaded, and where*

[29] http://validator.xml-cml.org/
[30] https://bitbucket.org/sea36/lensfield2/
[31] http://maven.apache.org/
[32] http://www.dropbox.com/

permitted, downloaded by anyone.

We do not expect a single solution to cover everything, and the more emphasis on security, the more effort required. In this phase of Quixote, we are publishing our work to the whole world and do not expect problems of corruption or misappropriation. We have therefore relied on simple proven solutions such as RESTful systems. Some of this is covered in the semantic architecture paper in this issue, and here we simply illustrate that initial systems at Cambridge have been implemented with AtomPub[33]*. Because the academic repository system has invested effort in the SWORD system*[34] *(which runs over AtomPub), this allows us to deposit/upload aggregations of files.*

Chempound repository

Quixote is built on CML compchem and, in our system, is further transformed to provide RDF used for accessing subcomponents and expressing searches. The Chempound (chem#) repository system[35] *(see Figure 3) has been built to support this. We expect that the first wave of distributed repositories will be using Chempound, and a publically accessible prototype repository is already in use within the Quixote project*[36]

Institutional repositories, DSpace

Institutional repositories (running software such as DSpace[37] *or Fedora*[38]*) may be responsible for storing the raw output files that are transformed into CML by the JUMBO-Converters. Alongside, they will also store basic metadata (authorship, usage rights, related works, etc.).*

This usage of institutional repositories distributes data management responsibilities among the institutions where the creators of the raw output files work. This provides an efficient basic data management support to the creators, and lets topic-specific repositories (such as Quixote's chem#) to focus on leveraging the specialized CML semantics extracted from the raw files, while still linking back to the original raw files at the institutional repositories. This schema also favors re-use of the same primary data by different specialized research topic repositories.

Yet antother temporary advantage of this approach is that, as the data collection increases, resource discoverability becomes a real challenge – even for the researcher herself. Even if much data can be extracted from the datafiles, some title and description metadata could be very useful to issue searches and can be

[33] http://tools.ietf.org/html/rfc5023
[34] http://www.ukoln.ac.uk/repositories/digirep/index/SWORD
[35] https://bitbucket.org/chempound/
[36] http:/quixote.ch.cam.ac.uk/
[37] http://www.dspace.org/
[38] http://fedora-commons.org/

provided by the person submitting the files to the repository. In the development phase, other researchers – as well as the dataset creator – would be able to discover and access a given unprocessed dataset without needing to wait for it to get processed and transferred into the final Chempound data repository.
Designing a DSpace-based raw data repository will also allow for defining a de facto *standardized metadata collection for compchem data description that may be very useful for harmonisation of data description in this specific research area – and might eventually evolve into some kind of standard for the discipline.*
At the present stage, we have done some preliminary work along metadata collection definition. A set of metadata has been defined and is being discussed in order to provide thorough descriptions of raw compchem datasets (potentially extendable to data from other research areas). Once the metadata set for bibliographical description of raw datasets is agreed, fields contained therein will be mapped to existing or new qualified DublinCore (QDC) metadata and a draft format will thus be defined. This format will be implemented at a DSpace-based repository, where trial-and-error storing loops with real datasets will be performed for metadata collection completion and fine-tuning – besides accounting for particular cases.

Avogadro

Avogadro is an open source, cross-platform desktop application to manipulate and visualize chemical data in 3D. It is available on all major operating systems, and uses Open Babel for much of its file input and output as well as basic forcefields and cheminformatics techniques. Avogadro was already capable of downloading chemical structures from the NIH structure resolver service, editing structures and optimizing those structures.
Input generation from these structures is present for many of the major computational chemistry codes Quixote targets such as GAMESS(US), GAMESS-UK, Gaussian, NWChem, MOPAC and others. These dialogs allow the user to change input parameters before producing input files to be run by the code. The output files from several of these codes can also be read directly, this functionality was recently split out into OpenQube – a library to read quantum computational code log files, and calculate molecular orbitals, electron density and other output.
Ultimately, much of this functionality will move into the Quixote parsers, with the OpenQube library concentrating on multithreaded calculation of electronic structure parameters. A native CML reader plugin has also been developed for Avogadro, to read in CML files directly and display the tree structure allowing visual exploration of CML files. As JUMBO and other tools can extract electronic structure, spectra and vibrational data, this plugin is being developed to extract them from the CML document.

Avogadro is already network aware, with a network fetch extension interacting with the NIH structure resolver and the Protein Data Bank (PDB). Experimental support for interacting with a local queue manager is also being actively developed, sending input files to the queue manager, and retrieving log files one the calculation is complete. Some data management features are being added, and as Chempound has a web API a plugin for upload, searching and downloading of structures will be added. A MongoDB-based application has been prototyped, using a document store approach to storing chemical data. This approach coupled with Chempound repositories and seamless integration in the GUI will significantly lower barriers for both deposition and retrieval of relevant computational chemistry output.

Avogadro forms a central part of the computational chemistry workflow, but is in desparate need of high quality chemical data. The data available from existing online chemical repositories is a good start, but having high quality, discoverable computational chemistry output would significantly improve efficiency in the field. Widespread access to optimized chemical structures using high level theories and large basis sets would benefit everyone from teaching right through to academic research and industry.

Results and Discussion

The Quixote project can manage input and output from any of the main compchem packages including plane-wave and solid-state approaches. The amount of semantic information in the output files can vary from a relatively small amount of metadata for indexing to a complete representation of every information output in the logfile. The community can decide at which point on the spectrum it wishes to extract information and can also retrospectively enhance this by running improved parsers and converters over the archived logfiles and output files.

The current test datasets in the Murray-Rust group are generated by parsing existing logfiles into CML using the JUMBO-Converters software. The amount of detail depends at the moment on the amount of effort that has been put into the parser. The current project is working hard to ensure inter-operability of dictionary terms and concepts by collating a top-level dictionary resource. When this is complete, the files will be re-parsed to reflect the standard semantics.

In the first pass, with the per-code parsers, we have been able to get a high conversion rate and a large number of semantic concepts from the most developed parsers. The use cases below represent work to date showing that the approach is highly tractable and can be expected to scale across all types of compchem output and types of calculation.

A typical final CML document (heavily truncated for brevity) is shown in Appendix 3. This whows the

structure of jobs and the typical fields to be found in most calculations.

Test dataset 1

The first use case consisted of 1095 files in Gaussian logfile format contributed by Dr. Anna Croft of the University of Bangor. These were deliberately sent without any human description with the challenge that we could use machine methods to determine their scope and motivation. We have applied the JUMBO-Parser to these, of which all except 5 converted without problems. The average time for conversion was between 3-10 seconds depending on the size of file. These files have now been indexed, mainly from the information in the archive section of the logfile but also with the initial starting geometry and control information. A large number of the files appear to be a systematic study of the attack by halogen radicals on aromatic nuclei.

Test dataset 2

This use case comprised of over 5000 files which Henry Rzepa and collaborators have produced over the years and which have been stored Openly in the Imperial College repository (helix). They are much more varied than the Croft sample and include studies on Möbius computational chemistry, transitional metal complexes and transition state geometries. A considerable proportion of the files emanate from student projects, many of which tackle hitherto novel chemical problems. It is our intention to create a machine-readable catalogues of these files and to determine from first principles their content and, where possible, their intent.

Test dataset 3

The NWChem distribution (NWChem-6.0) contains a directory (`/QA/tests/`) with a large number (212) of varied quality assurance tests for the software. All except 18 of these have been converted satisfactorily. One problem encountered was that the parser had used a large number of regexes which, when concatenated, scaled exponentially, so that some of the conversions took over a minute. We are now re-writing the parser to use linear time methods. These files cover a wider range of chemistry than the Croft and Rzepa contributions, as many of them use plane-wave calculations on solid state problems.

Test dataset 4

In the group of Pablo Echenique, at the Institute of Physical Chemistry "Rocasolano" (CSIC) and the University of Zaragoza, a large number of calculations were performed in peptide systems using the Gaussian quantum chemistry package. These calculations represent an exhaustive study (whose results and aims have been discussed elsewhere [14]), of more than 250 ab initio *potential energy surfaces (PESs) of the model dipeptide HCO-L-Ala-NH_2. The model chemistries investigated are constructed as homo- and heterolevels involving possibly different RHF and MP2 calculations for the geometry and the energy. The basis sets used belong to a sample of 39 representants from Pople's split-valence families, ranging from the small 3-21G to the large 6-311++G(2df,2pd). The conformational space of this molecule is scanned by defining a regular 12×12 grid from -165° to 165° in 30° steps in the 2D space spanned by its Ramachandran angles ϕ and ψ. This totals more than 35000 Gaussian logfiles, all generated at the standard level of verbosity, some of them corresponding to single-point energy calculations, some of them to energy optimizations. The use of JUMBO-converters through Lensfield2 has allowed to parse the totality of these files, through a complicated folder tree, generating the corresponding raw XML and structured compchem CML with a very high rate of captured concepts. The total time required to do the parsing was about five hours in an iMac desktop machine with a 2.66 GHz Intel Core 2 Duo processor, and 4 GB of RAM memory, running the Mac OS X 10.6.7 operating system.*

Quixote repository at Cambridge

The first repository (Figure 3) has been built at Cambridge (http://quixote.ch.cam.ac.uk) and has been viewable and searchable. In the spirit of Quixote this is not intended to be a central permanent resource but one of many repositories. It is available for an indefinite time as a demonstration of the power and flexibility of the system but not set up as a permanent "archive". It may be possible to couple such repositories to more conventional archive-oriented repositories which act as back-end storage and preservation.

Conclusions

Each day, countless calculations are run by thousands of computational chemistry researchers around the world, on everything from ageing, dusty desktops to the most powerful supercomputers on the planet.
It might be supposed that this would lead to a deluge of valuable data, but the surprising fact remains that most of this data, if it is archived at all, usually lies hidden away on hard disks or buried on tape backups;

often lost to the original researcher and never seen by the wider chemistry community at all.
However, it is widely accepted that if the results of all these calculations were publicly accessible it would be extremely valuable as it would:

- *avoid the costly duplication of results,*
- *allow different codes to be easily validated and benchmarked,*
- *provide the data required for the development of new methods,*
- *provide a valuable resource for data mining,*
- *provide an easy, automated way of generating and archiving supporting information for publications.*

In the rare cases when data is made openly available, the output of calculations are inevitably produced in a code-specific format; there being no currently accepted output standard. This means that interpreting or reusing the data requires knowledge of the code, or the use of specific software that understands the output.
A standard semantic format will:

- *allow tools, (*e.g. *GUIs) to operate on the input and output of any code supporting the format, vastly increasing their utility and range,*
- *enable different codes to interoperate to create complex workflows,*
- *additionally, if a semantic model underlies the format, data can easily be validated.*

The benefits of a common data standard and results databases are obvious, but several previous efforts have failed to address them, largely because of an inability to settle on a data standard or provide any useful tools that would make it worthwhile for code developers to expend the time to make their codes compatible.
The Quixote project aims to tackle both of these problems in a pragmatic way, building an infrastructure that can be used to both archive and search calculations on a local hard-drive, or expose the data on publicly accessible servers to make it available to the wider community.
The vision with which we started the Quixote project some months ago is one in which all data generated in computational QC research projects is used with maximal efficiency, is immediately made available online and aggregated into global search indexes; a vision in which no work is duplicated by researchers and everyone can get an overall picture of what has been calculated for a given system, for a given scientific question, in a matter of minutes; a vision in which all players collaborate to achieve maximum

interoperability between the different stages of the scientific process of discovery, in which commonly agreed, semantically rich formats are used, and all publications expose the data as readable and reusable supplementary material, thus enforcing reproducibility of the results; a vision in which good practices are wide spread in the community, and the greatest benefit is earned from the effort invested by everyone working in the field.

With the prototype presented in this article, which has been validated by real use cases, we believe this vision is beginning to be accomplished.

The methodological approach in Quixote is novel: The data standard will be consolidated around the tools and encourage its adoption by providing code and tool developers with an obvious reason for adopting the data standard; the "If you build it, they will come" approach. The project is rooted in the belief that scientific codes and data should be "Open", and we are therefore focussing our efforts on using existing Open Source solutions and standards where possible, and then developing any additional tools within the project. The Quixote project is itself completely Open, de-centralised and community-driven. It is composed of passionate researchers from around the globe that are happy to collaborate with anyone who shares our aims.

Authors' contributions

S. Adams has participated in the design of the Quixote system, is the main developer of Chempound and collaborated in the development of the compchem dictionaries and conventions.

P. de Castro has written the manuscript, and collaborated in the design of the D-Space-based soluion for metadata.

P. Echenique has written the manuscript, participated in the design of the Quixote system and help develop some of the tools contained in it.

J. Estrada has written the manuscript, participated in the design of the Quixote system and help develop some of the tools contained in it.

M. Hanwell has written the manuscript, participated in the design of the Quixote system and is a core developer of Avogadro.

P. Murray-Rust has written the manuscript, participated in the design of the Quixote system and he has been the main developer of the software tools.

P. Sherwood has written the manuscript, and collaborated in the design of the Quixote system.

J. Thomas has written the manuscript, participated in the design of the Quixote system and help develop

some of the tools contained in it.

J. Townsend has participated in the design of the Quixote system, developed the CML validator and collaborated in the development of the compchem dictionaries and conventions.

Acknowledgements

We thank all the many researchers that have contributed to the work discussed here with their ideas, testing and support; particularly Egon Willihagen, Anna Croft, Henry Rzepa, Lance Westerhoff, Luis Martínez-Uribe, Tamás Beke, Valera Veryazov, Weerapong Phadungsukanan, José Luis Alonso, Fermín Serrano, Isabel Bernal, and, of the library of Universidad de Zaragoza, Roberto Soriano, Miguel Martín, Teresa Muñoz and Ramón Abad (director). We also thank the ZCAM, and especially its Director, Michel Mareschal, for hosting and co-organizing the vibrant workshop in which the Quixote project was born. We thank as well the Daresbury Laboratory of the UK Science and Technology Facilities Council (STFC), which sponsored the First Quixote Conference, and EPSRC for supporting for the contribution of Jens Thomas through the Service Level Agreement with STFC. Both ZCAM and Daresbury are nodes of CECAM. Finally, thanks to Charlotte Bolton for the careful editing of the manuscript.

P. Echenique aknowledges support from the research grants E24/3 (DGA, Spain), FIS2009-13364-C02-01 (MICINN, Spain). P. Echenique and J. Estrada aknowledge support from the research grant 200980I064 (CSIC, Spain), and and ARAID and Ibercaja grant for young researchers (Spain). The mentioned meeting has been funded by ZCAM, the University of Zaragoza, Piregrid, the Aragón Government, and the Spanish Ministry of Science and Innovation. P. de Castro aknowledges support by Joint Information Systems Committee (JISC). Peter Murray-Rust acknowledges funding from JISC (CLaRION, XYZ) and EPSRC (Pathways to Impact).

References

1. *Frisch MJ, et al.:* ***Gaussian 03, Revision C.02****. [Gaussian, Inc., Wallingford, CT, 2004].*
2. *Gordon MW M S ans Schmidt:* ***Advances in electronic structure theory: GAMESS a decade later****. In* Theory and Applications of Computational Chemistry: The first forty years*. Edited by Dykstra CE, Frenking G, Kim KS, Scuseria, Amsterdam: Elsevier 2005:1167–1189.*
3. *Guest MF, Bush IJ, Van Dam HJJ, Sherwood P, Thomas JMH, Van Lenthe JH, Havenith RWA, Kendrick J:* ***The GAMESS-UK electronic structure package: algorithms, developments and applications****.* Molecular Physics *2005,* ***103****(6):719–747.*
4. *Valiev M, Bylaska EJ, Govind N, Kowalski K, Straatsma TP, van Dam HJJ, Wang D, Nieplocha J, Apra E, Windus TL, de Jong WA:* ***NWChem: a comprehensive and scalable open-source solution for large scale molecular simulations****.* Comput. Phys. Commun. *2010,* ***181****:1477.*

5. *Karlström G, Lindh R, Malmqvist PA, Roos BO, Ryde U, Veryazov V, Widmark PO, Cossi M, Schimmelpfennig B, Neograd y P, Seijo L:* ***MOLCAS: A program package for computational chemistry.*** Comp. Mat. Sci. *2003,* ***28****:222.*
6. *Echenique P, Alonso JL:* ***A mathematical and computational review of Hartree-Fock SCF methods in Quantum Chemistry.*** Mol. Phys. *2007,* ***105****:3057–3098.*
7. *Shao Y, et al.:* ***Advances in methods and algorithms in a modern quantum chemistry program package.*** Phys. Chem. Chem. Phys. *2006,* ***8****:3172–3191.*
8. *e-SciDR:* ***Towards a European e-Infrastructure for e-Science digital repositories - Final Report.*** http://e-scidr.eu/ *2008.*
9. *ESF:* ***European Computational Science Forum: The "Lince Initiative": from computers to scientific excellence*** *2009.*
10. *Harding ME, Metzroth T, Gauss J, Auer AA:* ***Parallel calculation of CCSD and CCSD(T) analytic first and second derivatives.*** J. Chem. Theory Comput. *2008,* ***4****:64–74.*
11. *Jensen F:* Introduction to Computational Chemistry. *Chichester: John Wiley & Sons 1998.*
12. *Szabo A, Ostlund NS:* Modern Quantum Chemistry: Introduced to Advanced Electronic Structure Theory. *New York: Dover Publications 1996.*
13. *Pople JA:* ***Nobel lecture: Quantum chemical models.*** Rev. Mod. Phys. *1999,* ***71****:1267–1274.*
14. *Echenique P, Alonso JL:* ***Efficient model chemistries for peptides. I. General framework and a study of the heterolevel approximation in RHF and MP2 with Pople split-valence basis sets.*** J. Comput. Chem. *2008,* ***29****:1408–1422.*
15. *Perczel A, Jákli I, Csizmadia IG:* ***Intrinsically stable secondary structure elements of proteins: A comprehensive study of folding units of proteins by computation and by analysis of data determined by X-ray crystallography.*** Chem. Eur. J. *2003,* ***9****:5332–5342.*
16. *Perczel A, Hudáky P, Füzéry AK, Csizmadia IG:* ***Stability issues of covalently and noncovalently bonded peptide subunits.*** J. Comput. Chem. *2004,* ***25****:1084–1100.*
17. *Cancès E, DeFranceschi M, Kutzelnigg W, Le Bris C, Maday Y:* ***Computational quantum chemistry: A primer.*** *In* Handbook of numerical analysis. Volume X: Special volume: Computational chemistry. *Edited by Ciarlet P, Le Bris C, Elsevier 2003:3–270.*
18. *Feller D:* ***The role of databases in support of Computational Chemistry.*** J. Comput. Chem. *1996,* ***13****:1571–1586.*
19. *Li L, Zhang R, Chen J, Zhang Y, Li L, Zhao Z:* ***ChemDataBase 2: An enhanced chemical database management system for virtual screening.*** *In* The Fifth Annual ChinaGrid Conference *2010.*
20. *Řezáč J, Jurečka P, Riley KE, Černý J, Valdes H, Pluháčková K, Berka K, Řezác T, Pitoňák M, Vondrášek J, Hobza P:* ***Quantum chemical benchmark energy and geometry database for molecular clusters and complex molecular systems (www.begdb.com): A users manual and examples.*** Collect. Czech. Chem. Commun. *2008,* ***73****:1261–1270.*
21. *Giese TJ, Gregersen BA, Liu Y, Nam K, Mayaan E, Moser A, Range K, Nieto Faza O, Silva Lopez C, Rodriguez de Lera A, Schaftenaar G, Lopez X, Lee TS, Karypis G, York DM:* ***QCRNA 1.0: A database of quantum calculations for RNA catalysis.*** J. Mol. Graph. Model. *2006,* ***25****:423–433.*
22. *Kotochigova S, Levine ZH, Shirley EL, Stiles MD, Clark CW:* ***Local-density-functional calculations of the energy of atoms.*** Phys. Rev. A *1997,* ***55****:191–199.*
23. *Murray-Rust P, Rzepa HS:* ***Chemical markup, XML, and the Worldwide Web. 1. Basic principles.*** J. Chem. Inf. Comput. Sci. *1999,* ***39****:928–942.*
24. *Guha R, Howard MT, Hutchison GR, Murray-Rust P, Rzepa HS, Steinbeck C, Wegner J, Willighagen EL:* ***The Blue Obelisk – Interoperability in Chemical Informatics.*** J. Chem. Inf. Model. *2006,* ***46****:991–998.*
25. *Schuchardt KL, Didier BT, Elsethagen T, Sun L, Gurumoorthi V, Chase J, Li J, Windus TL:* ***Basis Set Exchange: A community database for computational sciences.*** J. Chem. Inf. Model. *2007,* ***47****:1045–1052.*
26. *O'Boyle NM, Tenderholt AL, Langner KM:* ***cclib: a library for package-independent computational chemistry algorithms.*** J. Comput. Chem. *2008,* ***29****:839–845.*

27. *Steinbeck C, Han Y, Kuhn S, Horlacher O, Luttmann E, Willighagen EL:* ***The Chemistry Development Kit (CDK): An open-source Java library for chemo- and bioinformatics***. J. Chem. Inf. Comput. Sci. *2003,* ***43****:493–500.*

Figures

Figure 1 - Quixote architecture and conversion workflow

The user instructs Lensfield2 to convert output files of different computational chemistry codes into semantically rich CML files. The conversion is performed by JUMBO-Converters following the hints provided in the dictionaries and templates. The generated CML files are then transferred to one or more local and remote chem# repositories using a RESTful web API. The user can search and browse those repositories with a web browser, and can also manipulate and visualize the CML files with Avogadro.

Figure 2 - Quixote distributed repositories

A schematic view of distributed Quixote repositories. Some repositories push documents to the public web, others aggregate from it. There is (deliberately) no check on whether repositories have identical documents. Users can build search strategies that look for individual entries with specific data or make collections of documents that share or contrast properties.

Figure 3 - Chempound repository graphical interface

Chempound accepts either converted compchem CML or logfiles (which are then parsed by the JUMBO converters into compchem CML). The entries are indexed on 4 main criteria: (I) environment (program, host, dates, etc.) (II) initialization (molecular structure, basis sets, methods, algorithms, parameters, etc.) (III) calculation (the progression of optimization) (IV) finalization (molecular structure, properties, times, etc.) (a) Each entry is displayed with a thumbnail and key metadata (b) Properties and parameters for each entry, all searchable through SPARQL endpoint.

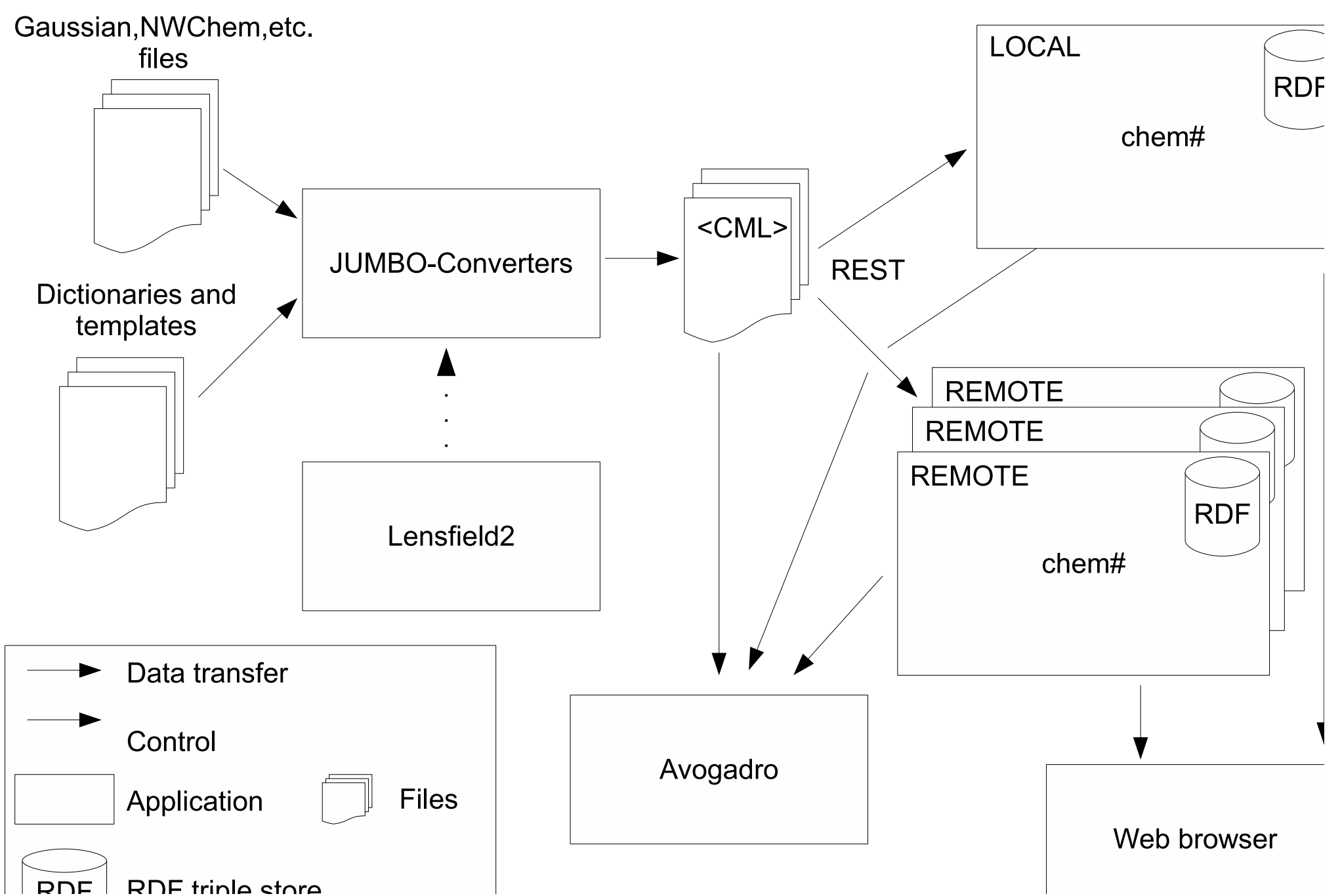

Gaussian,NWChem,etc.
files
Dictionaries and
templates
JUMBO-Converters
Lensfield2
<CML>
REST
LOCAL
RDF
chem#
REMOTE
REMOTE
REMOTE
RDF
chem#
Avogadro
Web browser
Data transfer
Control
Application
Files
RDF
RDF triple store

Figure 1

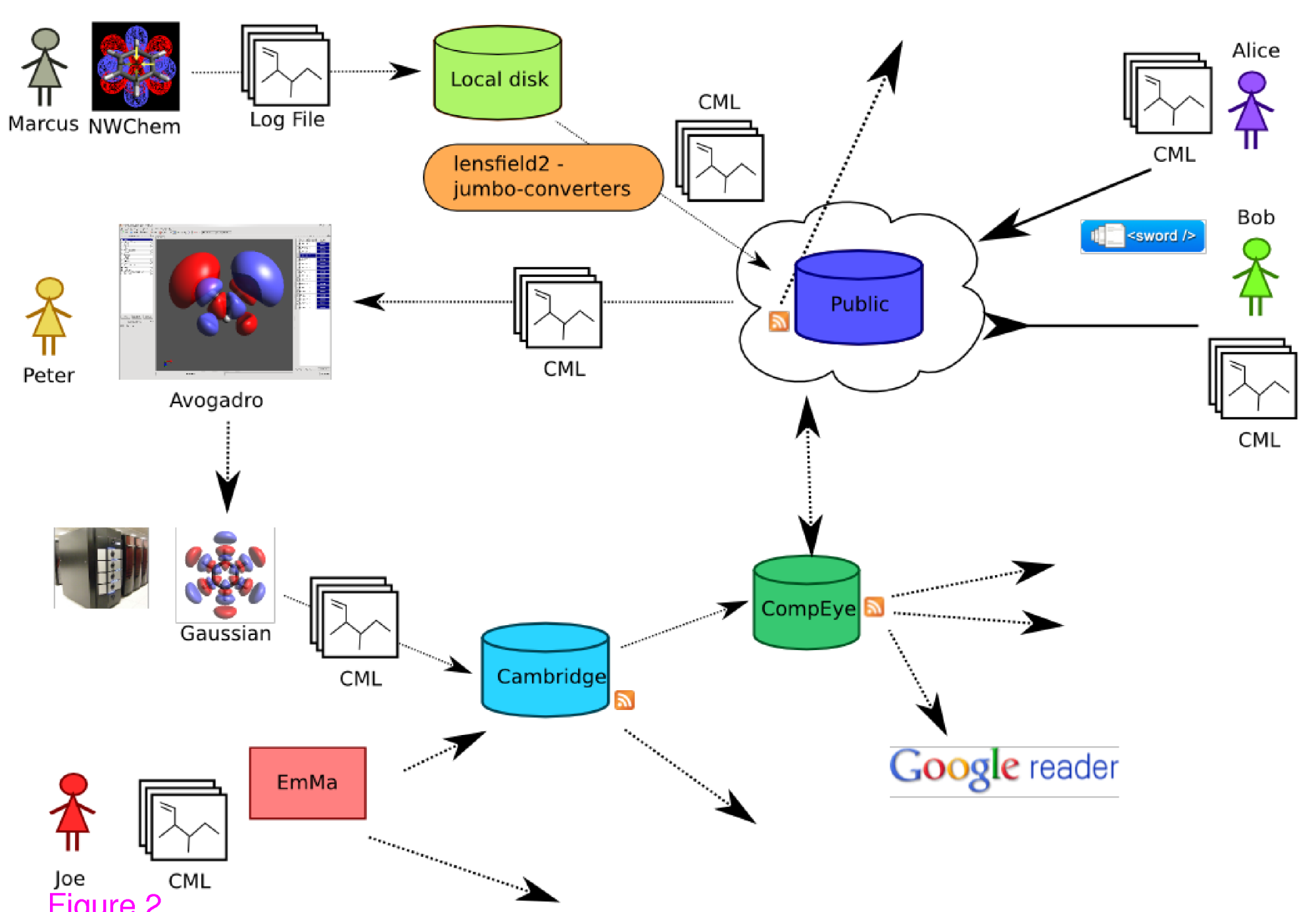

Marcus
NWChem
Log File
Local disk
CML
lensfield2 -
jumbo-converters
Alice
CML
Bob
<sword />
Public
Peter
Avogadro
CML
CML
Gaussian
CML
Cambridge
CompEye
Google reader
EmMa
Joe
CML

Figure 2

http://daedal.ch.cam.ac....

daedal.ch.cam.ac.uk:8080/content/index.html

chempound data repository

Home | Browse | Search | SPARQL | Feeds

Gaussian Calculation

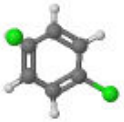
CH4

Gaussian Calculation

Cl

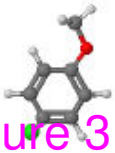
Gaussian Calculation

CH3



Gaussian Calculation

chlorobz/cl complex basis



Gaussian Calculation

Gaussian Calculation

Figure 3

http://daedal.ch.cam.ac....

daedal.ch.cam.ac.uk:8080/content/output.log_15/index.html

chempound data repository

Home | Browse | Search | SPARQL | Feeds

nitrobz/cl basis

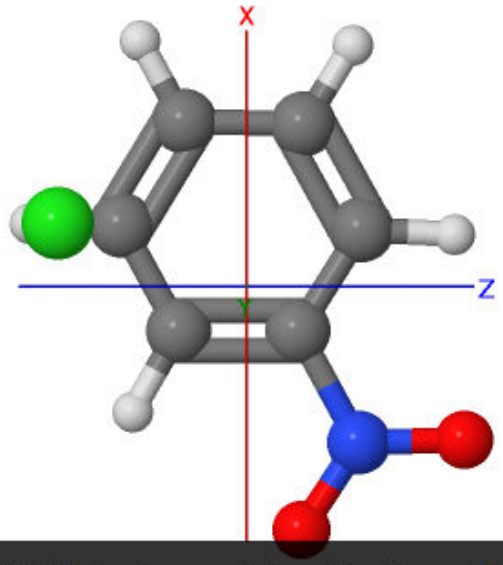
http://daedal.ch.cam.ac.uk:8080/content/output.log_15/

Calculation

Package	Gaussian 03 (Sun64-SVR4-Unix-G03RevB.04)
Method	UB3LYP
Basis Set	6-311+G(d,p)

Molecule

Formula	C6H5ClNO2(2)
Point Group	C1
Electronic State	2-A



Calculated Properties

Hartree-Fock Energy	-897.052

Figure 4

Additional files provided with this submission:

Additional file 1: appendix1.tex, 16K
http://www.jcheminf.com/imedia/2694512956638959/supp1.tex
Additional file 2: appendix2.tex, 2K
http://www.jcheminf.com/imedia/1395518328566389/supp2.tex
Additional file 3: appendix3.tex, 18K
http://www.jcheminf.com/imedia/4245891935663896/supp3.tex