Imperial College London

Research Data Management 'Green Shoots' Pilot Programme, Final Reports

This document contains the final reports of the six projects that formed Imperial College's RDM "Green Shoots" Programme.

In 2014, the Vice-Provost, Research, approved an allocation of £100K for academically-driven projects to identify and generate exemplars of best practice in RDM, specifically frameworks and prototypes that would comply with key funder RDM policies and the College position.

The call for projects outlined that frameworks could be based either on original ideas or integrating existing solutions into the research process, improving its efficacy or the breadth of its usage. There was an expectation that solutions would support open access for data; solutions that supported Open Innovation were strongly encouraged.

Six projects were funded, covering different disciplines, faculties and research areas. The projects ran for six months, finishing at the end of 2014. This document contains the final reports of the following projects:

1. **Haystack – a computational molecular data notebook**, by *Clyde Fare* and *Michael Bearpark*
2. **Imperial College Healthcare Tissue Bank**, by *Geraldine Thomas*, *Sarah Butcher* and *Christopher Tomlinson*
3. **Integrated Rule-based Data Management System for Genome Sequencing Data**, by *Michael Mueller*, *Simon Burbidge*, *Steven Lawlor* and *Jorge Ferrer*
4. **Research data management in Computational and Experimental Molecular Sciences**, by *Henry S. Rzepa*, *Matt Harvey*, *Andrew Mclean* and *Nick Mason*
5. **Research data management: Where software meets data**, by *Christian T. Jacobs, Alexandros Avdis, Gerard J. Gorman and Matthew D. Piggott*
6. **Time Series RDM Green Shoots Report**, by *Nick Jones*

The Green Shoots Programme was managed by Ian McArdle (http://orcid.org/0000-0002-2221-8866) and Torsten Reimer (http://orcid.org/0000-0001-8357-9422), Research Office.

Imperial College London, 2015

Haystack
– a computational molecular data notebook

Clyde Fare and Michael Bearpark

Within computational chemistry, sharing and reuse of data currently lags behind other research fields despite the significant benefit of open data to the advancement of science. In this summer project we extended a working prototype of a computational chemical notebook, making it available for all on github[14]. This notebook enables computational molecular researchers to easily share a curated subset of their results and document how those results were generated.

Context

Research councils increasingly require research data to be made available for projects they fund. Similarly some journals[1] now require authors to make their research data available. This push has led to the development of platforms enabling archiving and sharing of data [2]. There have been attempts to make the codes used in scientific research more open and accessible so that the accuracy of their calculations can be evaluated [3,4] alongside experiments with open-notebook science [5], where all research output is made available as it happens. In the field of computational chemistry the culture of sharing data is not yet established although the Quixote project [8] – an initiative to add metadata and archive calculations to web-accessible databases – is under way.

One of the things missing from the theoretical chemical tool set is a means of including a curated part of the actual research process alongside the published results. A painless way of capturing the process used to create the data, plots and analysis etc. (such that it could be made available alongside a published article without requiring significant additional effort from the authors) would greatly increase the transparency and reproducibility of the published research literature.

In an attempt to work towards this goal we built upon the IPython Notebook [6,7] – a computational notebook based on the Python programming language and two prominent python based electronic structure frameworks: the atomic simulation environment[10] and cclib[11]. Our prototype computational chemical notebook added the following functionality:

- Calculations using mainstream computational chemistry software can be set up.
- Calculations can be submitted to run on a high-performance computing cluster.
- Data from completed calculations can be retrieved and visualised.

The prototype was presented at the PyData 2014 conference ("*Changing the way scientists, engineers, and analysts perceive big data"*) and the Thomas Young Centre for materials simulation and at Euro Scientific Python 2014.

Problems

In order for our prototype to be useful to a wider audience it required modification to make it general to computational chemistry rather than specific to the individual setup used by the Bearpark group. It also required updating to keep up with changes in its parent project the IPython notebook. The prototype was based on numerous scientific libraries making installation a lengthy and somewhat challenging process; this overhead represented a significant barrier to adoption. A further problem was the inability of a single notebook 'page' to capture all the research that occurs in a project. Thus a means of connecting and sharing a collection of notebook pages in a sensible manner was needed.

Approach

The aforementioned issues separated into three domains: refactoring and updating the prototype, testing to weed out bugs, and adding a project tree feature.

For refactoring we sought to separate functionality that should be general e.g. interfacing with computational clusters, from code specific to our particular choice of quantum chemical package. This would allow others to use their own quantum chemical codes within the notebook. We also aimed to update the prototype to use the latest version of IPython and the molecular visualisation tool to its successor JSMol[12] to solve browser compatibility issues. Finally we aimed to make use of commonly available open source tools[13] to make our code and all its dependencies easy to install on any of the three major operating systems so that anyone with an internet connection could easily and freely obtain and make use of it.

Our main feature enabled by the latest version of IPython was the construction of a means to keep track of an entire project composed of multiple notebook pages. We imagined a project 'tree' (i.e. a linked set of nodes) where each node in the tree corresponded to a notebook page containing a set of calculations and their analysis. The entirety of research involved in a project would be contained within this tree. This project tree would be implemented as an alternative notebook dashboard.

For testing, a UROP student was tasked with completing a simple quantum chemical project using the notebook to annotate and keep track of the calculations he performed. He was also to explore the features of the notebook proposed above and test installation on different systems.

Achievements

We successfully completed updating of our notebook code to use the latest IPython 2.x base and our molecular visualiser to use the JSMol package. We refactored the code allowing calculations performed with the popular Gaussian quantum chemistry package such that it was independent of our particular computational resources and could be used by anyone with access to Gaussian. We further removed dependence of the notebook code on our Gaussian interface allowing any of the many quantum chemistry packages with interfaces defined in the atomic simulation environment to be used.

A project tree view was implemented allowing a collection of notebook pages to be linked together to represent a project. Further a means of selecting collections of nodes and archiving them for inclusion in a publication was created. Installation was streamlined by constructing recipes using the

conda build environment popular in the scientific python community. This makes installing all of the necessary libraries our code depends upon automatic, allowing easy installation on all operating systems. Finally our code base has been made available via github and has a BSD license.

Our UROP student successfully undertook an undergraduate project using the computational chemical notebook. And both a PhD student and a new Msci student are making heavy use of the computational notebook.

During the project period we presented a lightning talk at the scientific python 2014 conference.

Lessons learned

In terms of the development process the technical debt incurred during the construction of the prototype took longer than anticipated to pay off. This is a general feature of software development where a higher pace of development leads to code that is not robust and hence a time debt that needs to be paid to rewrite the code in a more robust form. In our case our prototype builds upon several other tools and libraries that were themselves in development. The rapid pace of development of these libraries slightly complicated our situation. A better approach would have been to decide to stick with particular versions of the libraries whilst refactoring our own code and then choose which libraries to update. Whilst in some way this means more code will need to be rewritten it would have simplified the process.

Observing the working habits of our UROP student during his undergraduate project we saw that there were multiple ways of making use of the chemical notebook. This meant unexpected choices were made, some of which were novel and meant he could perform calculations in an efficient manner. However, some meant the notebooks themselves no longer constituted reproducible research as he had stripped out of them the data needed to reproduce the calculations. The power of the computation tools available reinforces the need for a tight coupling between the tools available to make research reproducible and working habits that promote reproducible research.

A further point is the dependence of our project on understanding the basics of a programming language. Whilst usage of the notebook does not require a high degree of programming skill and the Python language itself is designed to be as readable as possible, some programming like activity is necessary consequently there is some overhead to using our tool.

References

1. PLOS one data policy. at <http://www.plosone.org/static/policies.action#sharing>

2. Figshare. at <http://figshare.com/>

3. Software Carpentry. at <http://software-carpentry.org/>

4. Mozilla Science Labs. at <http://mozillascience.org/>

5. Open Notebook Science. at <http://www.nature.com/news/2008/080915/full/455273a.html>

6. Pérez, F. & Granger, B. E. IPython Notebook. at <http://ipython.org/notebook>

7. Pérez, F. & Granger, B. E. IPython: a System for Interactive Scientific Computing. *Comput. Sci. Eng.* **9,** 21–29 (2007).

8. Adams, S. *et al.* The Quixote project: Collaborative and Open Quantum Chemistry data management in the Internet age. *J. Cheminform.* **3,** 38 (2011).

9. Alfred Sloan Foundation Grant. at <http://ipython.org/sloan-grant.html>

10. O'boyle, N. M., Tenderholt, A. L. & Langner, K. M. cclib: A library for package-independent computational chemistry algorithms. *J. Comput. Chem.* **29,** 839–845 (2008).

11. Bahn, S. R. & Jacobsen, K. W. An object-oriented scripting interface to a legacy electronic structure code. *Comput. Sci. Eng.* **4,** 56–66 (2002).

12. Hanson R et al. Jsmol and the next-generation of web-based representation of 3d molecular structure as applied to protopedia. *Israel J. chemistry* **53**, 207-216 (2013)

13. Oliphant, T, Anaconda at <http://conda.pydata.org/>

14. Fare, C cc_notebook at <https://github.com/Clyde-fare/cc_notebook>

Imperial College Healthcare Tissue Bank Green Shoots Grant Outcomes Report

Geraldine Thomas, Sarah Butcher and Christopher Tomlinson

Introduction

Tissue banks collect and store biological material in different formats obtained from healthy individuals or from patients. However, this material is only useful for research if associated with data. The richer the dataset that accompanies the sample, the more useful the sample is for research purposes. This data can include information on variables that affect the suitability of the material for different types of analysis, data on the phenotype of the disease, the individual, and increasingly the results of clinical tests, including genotypic information. Obtaining data on treatment and outcome can be challenging with a mobile population. A patient may start their journey in one NHS Trust and end it in another. The majority of NHS Trusts do not have data sharing agreements in place, which means negotiation with each Trust for access to data. One possible solution is to link to national databases that record longitudinal outcome data for individual patients, such as the National Cancer Registry Service.

One of the richest sources of data to enrich annotation of remaining samples from the same individual is often not collated – data generated from research use of samples. Most tissue banks provide analysis-ready samples for use by researchers (extracted nucleic acids, tissue sections) as a way of making a limited resource go further. This results in multiple research groups being provided with samples from the same patient, and even from the same piece of tissue. Tracking research data back to individual samples and individual patients is a challenge, but one that can provide a rich source of varied data for secondary use by bioinformaticians interested in developing algorithms for systems biology approaches to understanding disease processes.

The Imperial College Healthcare Tissue Bank (ICHTB) is a diverse collection of physical tissue specimens that have been collected from operations on patients in the hospitals trust. At the current time, there are approximately 60,000 available samples from around 15,000 different donors and over the past ten years more than 20,000 tissue bank samples have been issued to researchers. Information about the specimens is contained in an online database system. The system contains detailed, anonymised records about donors, operations and samples including pathology reports and allows searches of the collection to take place.

Objectives

In spring 2014, the ICHTB team was awarded a Green Shoots grant to augment the existing ICHTB database with further information that would sit alongside the patient and sample record. In our proposal we focussed on gathering data from recent experiments that have taken place on samples from the collection and recording this alongside additional information from the patient record. The integration of external information greatly enhances the utility of the samples in the collection.

Process

In the initial stages we carried out a survey to identify the types of experiments that are currently or typically carried out on samples from the ICHTB. This initial survey identified a wide range of different and specialised data types, each with a requirement for different associated formats and metadata. For the scope of this project we narrowed our search to look for data types that are routinely created from tissue bank samples, to act as an exemplar. This approach would have the most impact in the project lifetime, as an import pipeline for a popular data type could be reused every time data of that type is generated.

Within the Imperial Hospitals NHS trust, it is now policy to sequence tumour samples for certain types of cancer to aid diagnosis and treatment decisions. This approach is currently used for lung cancer samples and (we feel) is likely to be extended to other cancers in the short/medium term. Currently, a targeted sequencing gene panel approach is performed on the IonTorrent sequencing platform and after considerable analyses, a standard format report containing information about suspected mutated genes is produced by the laboratory staff. We were able to exploit existing contacts within the NHS trust to meet with the laboratory staff to gain access to sample data and to aid our understanding and interpretation of the current sequencing reports. It is anticipated that this working relationship will extend beyond the end of the RDM grant and into the foreseeable future.

Alongside the ongoing work on sequencing data, we were able to use the funds from the Green Shoots grant to fully implement a link with the National Cancer Registry. The purpose of this strand was to augment the existing donor information in the tissue bank with outcome information about the donors. Outcome information greatly enhances the utility of the samples for a given donor, particularly if the cause of death is recorded as it is in the NCR records. Collecting this information will enable researchers to examine genetic markers for survival rates in particular cancer types as those patients that have died of a condition can be separated from those that have survived, or have died from an unrelated cause.

Putting the two areas of work together will, at some future point, give the ICHTB a subset of donors whose outcome is known as well as sequencing information on samples taken from them. We believe that this will be of great use to the medical research community at (and beyond) Imperial College.

Results

We were able to first build a tool for automatically exchanging information with the National Cancer Registry and then to register 1884 of our donors with them. The tool automatically sends messages about available samples to the NCRS and receives and interprets the reply. The ICHTB database is updated with the outcome data where it is known about a patient. Primary and secondary causes of death are recorded where they are present and this information is presented back to the user alongside the donor record in the ICHTB user interface. Of the 1884 patients registered with the NCR we found that 572 were identified in their records. Of these 408 were still alive at the time of writing and 164 were dead – the rest of the set are currently unidentified by the NCR. This system is now actively in use and will continue to be used to update these links between ICHTB and NCR data.

Once the NCR data exchange tool was built and deployed, we turned our attention to the more complex problem of handling our prototype direct sequencing datasets. Each analysis of this type yields multiple levels of data, with differing degrees of useful ness for different audiences, also differing perceived levels of reuseability for future research. We discussed how we could best represent data arising from these analyses in a way that was useful in the ICHTB context. We received a representative data report type from the NHS trust laboratory and set about building a pipeline to import this information directly into the ICHTB, together with all relevant metadata required to track their provenance. A pipeline has been built and a prototype user interface to view these derived sequencing data has been constructed. As an adjunct, we have explored how raw data pertaining to the sequencing reports are currently stored (the aligned reads and raw non-filtered variant calls). These files can be large (multiple gigabytes) and should not be copied unnecessarily. In future, we expect to extend the ICTB interface to allow tracking of the raw data files themselves as they are stored and archived separately by the generating laboratory. This will increase the potential for later re-use. This work is still at the prototype stage and is not yet being used with real data. Work on this aspect of the ICHTB will continue beyond the end of the RDM funding period.

To assist with tracking data outcomes arising from ICHTB-associated projects, we also started to investigate ways of improving the linkage between the ICHTB and the research papers arising from ICHTB tissues used in approved research projects. It is a requirement that all approved projects report back to ICHTB and presently, publications are reported back periodically and made available by the ICHTB as a static list. We started to consider how these data could be more effectively linked into the data holdings associated with the ICHTB. Whilst publications can be easily linked to specific studies, each study may utilise many different samples from many different patients. Moreover, there is no trivial link between reported results and any single sample. We considered linking these papers into the university's research data repository Spiral. This proved to be complex for a number of reasons; including formulating the most appropriate mapping level between the publication and the groups of samples used; the fact that some of the papers are linked to external authorship and may not be present within Spiral and furthermore that Spiral has no publically accessible API that could be queried to make the link. Within the scope of the project, this line of work was not pursued further but will be a point for further work.

Impact

This project has established a working mechanism that enables linkage with the National Cancer Registry for all cancer patients who have donated material to the Imperial College Healthcare Tissue Bank (ICHTB www.imperial.ac.uk/tissuebank). The mechanism preserves patient confidentiality, and has been approved by the NHS Trust Caldicott Guardian. In addition, we have developed a pilot system to add data obtained from existing routine next generation sequencing to individual tissue samples held in the bank. This type of linking is becoming ever more important in light of a number of recent developments in this area, (such as Genomics England and other large initiatives) where samples and routine local screening and QC data arising from them may need to be routinely tracked with respect to results returned from externally run analyses.

Integrated Rule-based Data Management System for Genome Sequencing Data

Michael Mueller, Simon Burbidge, Steven Lawlor and Jorge Ferrer

Background

The advent and rapid evolution of next-generation DNA sequencing (NGS) technologies has revolutionised the field of genome research and has created new opportunities for the application of genomics in biomedical research and clinical practice. The ability to generate large amounts of information about individual genome sequences in a short amount of time at continuously falling costs has lead to a widespread adoption of NGS technologies. Consequently, research institutes are now faced with increasingly large volumes of sequencing and associated data requiring more sophisticated approaches to data storage and management then commonly in place to date in order to ensure data integrity and security, avoid unnecessary data replication and facilitate data access for analysis and sharing.

While the implementation of large-scale data management systems is a challenge biomedical research has been confronted with only recently other data intense domains such as physics have been developing solutions to problems associated with the storage, management and sharing of large data collections since more than two decades. More recently, the genomics research community has started to adopt existing technologies that originated in these fields. Leading this trend are large-scale sequencing centres such as the Wellcome Trust Sanger Institute in the UK or the Broad Institute in the US, where data grid middleware such as the integrated Rule-Oriented Data management System (iRODS) has been successfully used for storing and managing large distributed sequencing data sets and associated metadata.

Requirements

The newly established DNA sequencing service at the NIHR Imperial BRC Genomics Facility will generate large amounts of genome sequencing data for users across Imperial College. The Illumina HiSeq2500 DNA sequencer installed in the Facility will output around 10TB of raw data every two weeks amounting to a total data throughput of 260TB per year. Manual processing, analysis and dissemination of sequencing data make data management time consuming, pose risks to data integrity and may result in unnecessary data replication. To address these issues the Facility will set up a data management system for the new sequencing service that should integrate with existing HPC infrastructure for processing, analysis and dissemination of raw sequencing data and analysis results. The system should i) optimise data storage usage ii) facilitate data sharing within the college and with external collaborators iii) comply with College and funder data preservation and protection policies iv) allow for a high degree of automation and v) be scalable. Figure 1 shows an overview of the data life cycle within the proposed system.

iRODS

The implementation of the system is based on the iRODS middleware. iRODS is a widely used, scalable, open source data grid system that has been successfully adopted for the management of next-generation sequencing data at other research institutions such as the Wellcome Trust Sanger Centre, the Broad Institute and the University of Uppsala. The iRODS platform implements a logical name space providing transparent access to distributed and disparate storage resources. iRODS features important for the implementation of the system are the rule engine and the meta data catalogue. The rule engine allows the invocation of pre-defined sequences of action (micro-services) in regular intervals or when particular events occur. State changes resulting from rule execution are stored and this information can be used to track and control rule execution. The rule engine will facilitate the effective management of data life cycles (file migration, file format conversion etc.), data preservation (consistency, replication and archiving) and data security (encryption). The metadata function allows to associate descriptive system and user defined metadata with files. This will enable search, management and tracking of data and data manipulation within the system.

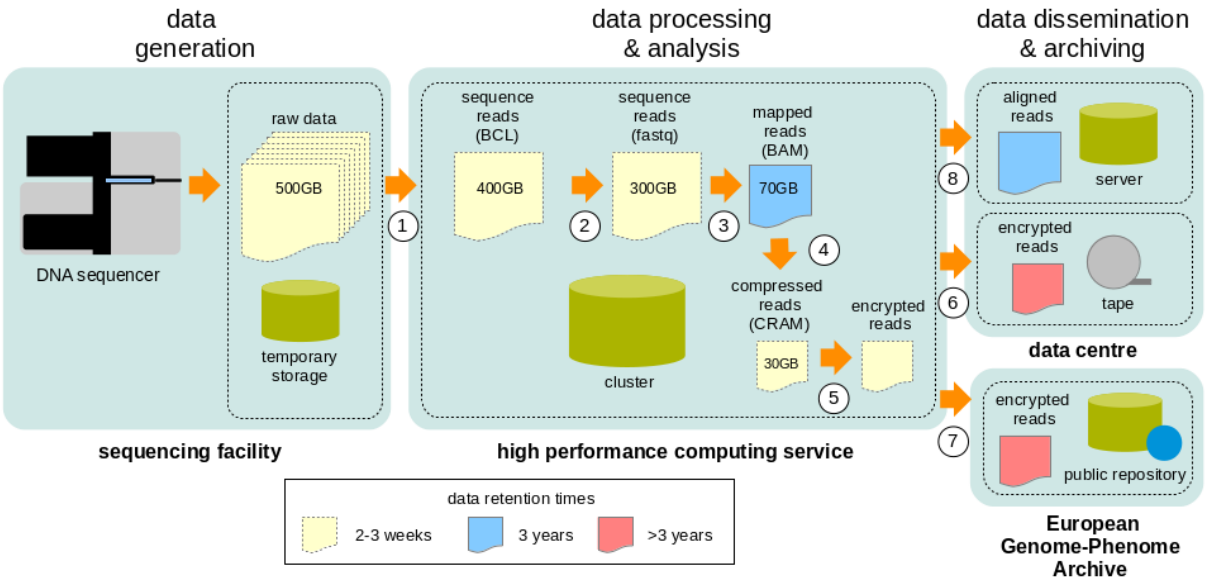


Figure 1 Data management system for genome sequencing data. Raw sequencing data is transferred from a local storage server at Hammersmith Campus to the HPC Service at South Kensington (1), read data is converted from the vendor specific BCL format to the platform independent fastq format (2), sequencing reads are mapped to a reference genome sequence (3), mapped read data is compressed further using reference based read compression (4), compressed reads data is enrypted (5), encrypted read data is archived on tape (6) and remotely at a public repository (7), aligned read data is disseminated locally to users (8).

Implementation

We have implemented a prototype of an iRODS based data management system that comprises all steps of the data processing required to disseminate sequencing data to internal users of the facility (steps 1 – 4 and 8 in Figure 1): (1) upon completion of a sequencing run raw data required for the conversion of raw sequencing data from the vendor specific BCL format to the platform independent fastq format is transferred to a storage server at the HPC Service in South Kensington. (2) Reads are

converted from BCL to fastq format using the vendor supplied bcl2fastq software. Splitting of read data by sample and project also occurs during this step. Subsequently a quality control report is generated with FastQC. (3) Reads in fastq format are then mapped to a reference genome sequence with BWA MEM which outputs mapping results in BAM format resulting in a >3-fold reduction in file size. (4) Further compression is achieved by applying a reference based compression algorithm using samtools v1.0 to convert BAM to CRAM files. (8) Aligned read data in BAM format is transferred to a webserver and made accessible for download through the web-based iRODS client iDrop Web 2.

The initial proposal envisaged management of the entire data processing workflow through the iRODS system. Figure 2A outlines the iRODS setup originally proposed with iRODS resource servers running on all storage resources. The webserver is also an iCAT enabled resource server that hosts the iRODS meta data catalogue. In this setup the entire data life cycle can be managed and tracked by iRODS.

However, concerns were raised by the HPC service team regarding user authentication and file ownership management. The default iRODS setup requires iRODS user authentication separately from the system authentication. Apart from the additional administrative overhead the trustworthiness of the iRODS authentication system was questioned by the HPC service team and considered a potential security risk. With regard to file ownership management, the fact that all files that are put into iRODS will be owned by a single user (the user the iRODS service is running as) was considered a problem as this creates complete dependence on the iRODS file ownership management.

Figure 2 Proposed vs implemented setup of the data management system. A) Proposed iRODS setup in which all storage resources in the system are iRODS resource servers belonging to the same iRODS 'zone' (igfZone). B) Implemented iRODS setup. Only storage resources administrated by the Genomics Facility run iRODS resource servers. The storage resources administrated by the HPC service are outside the iRODS system.

Firstly, to address the concerns regarding user authentication it was suggested to enable iRODS PAM authentication which allows the use of system passwords for iRODS authentication. Secondly, the issues relating to file ownership could be overcome by implementing the HPC iRODS resource as a 'direct access resource'. Direct access data resources are accessible both through iRODS and through the file system. iRODS acts as an "overlay" for the UNIX file system providing meta-data annotations for the files in the file system. However, in this operation mode iRODS runs as a root process which means that iRODS could potentially access/write any data on the system. The complete reliance on the authentication and access control in iRODS was not considered acceptable. Due to the concerns regarding data security and integrity the system was implemented without the integration of the HPC storage resource into iRODS (see Figure 2B). Instead data is fetched from iRODS via an iRODS client installed on the HPC system. After processing of the data on the cluster results and metadata are pushed back into iRODS.

Conclusion

We have successfully implemented a prototype of a data management system for sequencing data generated by the Imperial BRC Genomics Facility. The use of iRODS to manage and track the data life cycle within the system constitutes a compromise between the powerful functionality of iRODS and the sacrifice of file system based control over data access and ownership. This becomes an issue when shared resources such as HPC systems need to be integrated where iRODS based file management might conflict with the existing setup of the system. These issues would need to be addressed if the iRODS system was to be i) extended to manage data generated by secondary data analysis workflows and ii) used more widely across the college to manage sequencing and other research data.

Research data management in Computational and Experimental Molecular Sciences

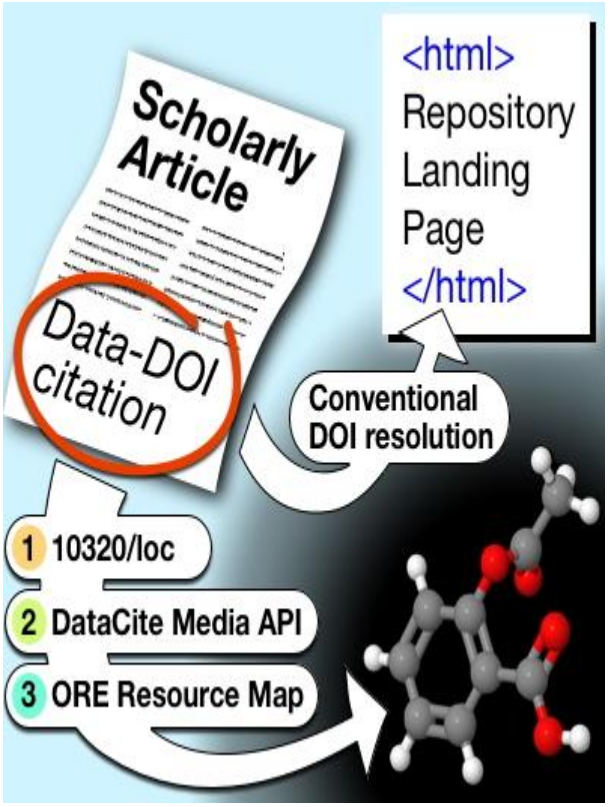
Henry S. Rzepa (ORCID: 0000-0002-8635-8390), Matt Harvey (ORCID: 0000-0003-1797-3186), Andrew Mclean and Nick Mason (ORCID: 0000-0001-9475-0328)

Context of the project

This project is based around an existing data repository (SPECTRa), its font end (uportal) and the enhancement of the system by enabling standards-based access to the data held on the repository, thereby allowing creation of visual representations of the data for use in tables and figures published in scholarly journals and incorporation into conference presentations and teaching material. An important element of this project has been the development and improvement of an overall workflow which provides a practical (and proven) method of data gathering, both new and already existing, together with the addition of the essential data and DOIs that actually facilitate subsequent discovery and re-use.

Issues

1. The first issue addressed was to package up a workflow based on the software systems constituting the uportal front end into a package that could be openly distributed for installation and use by other research groups, departments or institutions.
2. The data repository itself is based on DSpace, and we wished to enhance its functionality by adding a retrospective DataCite persistent identifier to each item in the repository, and to investigate the appropriate metadata for optimising the interaction between SPECTRa and various DataCite services.
3. In preliminary work prior to this project, we had implemented an extension to the DSpace Handle records to allow machine access to the individual data files based on specifying their persistent identifier together with a specification of the type of datafile required. This system was not directly compatible with the DataCite mechanisms providing this functionality, an issue that we addressed in the present project.
4. We wished to address the issue of data discovery and datametrics, based on metadata records and the facilities provided by DataCite.
5. The issue of content retrieval by authors wishing to make use of these features in journal articles.



6. The issue of facilitating data curation of existing data collections which did not conform to modern repository standards.
7. The issue of disseminating best repository and RDM practice and the experiences we had gained as a result of our own project.

Implementations and deliverables

These are discussed in more detail on our demonstrations page. A brief summary of the main points is below.

8. A professional programmer was employed to refactor the uportal front-end code and create an installer package that others can use to build their own repository system based on the DSpace open code.
9. The metadata held on SPECTRa has been added to and updated to make it fully compliant with the latest DataCite specifications. This involved adding cross-walks between the internal DSpace metadata schemas and those used by DataCite. The metadata includes full incorporation of the ORCID identifier.
10. Examples of data discoverability using the DataCite search resources have been collected.
11. We have added two further mechanisms to the machine actionable procedures available for automatic retrieval, visual presentation and re-use of data from the SPECTRa repository. These have been incorporated into a talk and demonstration (DOI2Data) intended for presentation at the FORCE2015 conference on Research communications and e-scholarship in January 2015.
12. The curation of a ten year old set from Cambridge University is now available as a SWORD-endpoint. The project is adding around 140,000 newly curated datasets and illustrating the use of critical discovery metadata.
13. A template based on the DOI2Data demonstrator has been deployed in three published high-profile journal articles. Two further articles are being prepared on the results of this project for peer-reviewed publication to encourage such activity by other authors. The template is also now incorporated into enhanced teaching notes for undergraduate lectures.
14. A presentation on these themes was given at the ODIN-2014 meeting in Amsterdam in September 2014.
15. Our project places Imperial 2nd of UK universities and the 7th highest datacentre globally in terms of external exposure, visibility and discoverability of open research data.

Lessons learnt

16. The importance of SEO or search engine optimisation using standards-based metadata deployment to enable data-discovery.
17. The unique local expertise in the procedures gained from the project will allow Imperial College to maintain its rank as a leading global site in RDM and data discovery and re-use.

Recommendations

18. With the SPECTRa data repository now upgraded to conform to the important DataCite metadata standards, we recommend that these features be considered for incorporation into both the institutional and any data repositories that may be deployed in the future.
19. External repositories such as Figshare/Arkivium either currently do not implement the features we have introduced, or do so only partially. We recommend that if such external repository service providers are considered for future use within College, these aspects be considered for inclusion in the operational requirement drafted as part of any procurement.
20. We encourage new RDM resources to be considered for incorporation into both undergraduate and postgraduate teaching as the means for introducing best practice in RDM at an early stage.
21. It is important to establish a culture of data curation and sharing in College. The provision of workflows of this type for other disciplines will help to overcome some of the impediments to sharing envisaged by even those researchers keen to share.

Research data management: Where software meets data

Christian T. Jacobs, Alexandros Avdis, Gerard J. Gorman and Matthew D. Piggott

Context

Computational science workflows are characterised by the interactions between both software and data. Collected data (e.g. tidal forcing conditions) is frequently given to scientific software (e.g. a numerical model) as input. This software then produces 'output data' (e.g. atmospheric flow fields or diagnostic quantities) which is analysed by the researcher to yield scientific results. In order to ensure reproducibility and recomputability of these results, the software, input data and output data should all be captured along with any provenance metadata. However, these components are often not published alongside the main scientific findings in journal articles.

This 'Green Shoots' RDM project investigated ways in which scientific software and data could be shared online at literally the 'push of a button' It aimed to facilitate their publication in online, citable and persistent repositories by introducing a large amount of automation, which in turn motivates researchers and encourages the sharing and re-use of research outputs through a more open and easy-to-use scientific workflow.

Project Deliverables

The main deliverable of this project has been the development and release (under the GNU General Public License) of an open-source software library, PyRDM (github.com/pyrdm). This library is able to automatically publish software source code (stored under Git version control) and data to online repositories provided by Figshare (https://figshare.com). In return, Figshare yields a Digital Object Identifier (DOI) for each repository which can be used in journal articles to formally and properly cite research outputs. For example, the specific version of the software used to produce a given dataset/result is often not stated in journal articles; instead, it is usually the software's website or generic user manual that is cited. PyRDM is able to automatically determine the particular version of the software being used and publish it to a Figshare repository. The DOI that is minted can then be used to reference that repository to enable much better recomputability of the data. Metadata is also added to the repository automatically. This includes author information determined from the software's AUTHORS file to achieve proper affiliation, and the software's version identifier; if another researcher tries to publish the same version of the software, then the Figshare repository and DOI are simply re-used instead.

In order to demonstrate its functionality, PyRDM has been integrated into the Fluidity computational fluid dynamics code (http://www.fluidity-project.org). Researchers can perform a fluid flow simulation, and then run a Fluidity-specific publishing tool which uses the PyRDM library. This tool automatically determines the version of the software used to run the simulation, and uploads it to Figshare. The relevant input and output files can also be published by providing a minimal amount of information, such as their location on the computer, in the simulation's configuration file. Throughout the publication process, the DOIs that are minted are stored in the configuration file and

the simulation's metadata to (a) improve data provenance and (b) enable a new revision of the repository to be created if the data is updated at a later stage.

A more detailed description of PyRDM, its functionality, and an example of its application has been published in the Journal of Open Research Software (see [3]).

Recommendations

Throughout the implementation and use of PyRDM, several issues have been identified. For example, attempting to automatically affiliate software authors to an online repository can prove difficult due to lack of standardisation. For example, all authors must currently provide their Figshare author IDs in order for PyRDM to correctly attribute them to the software repository on Figshare; for a different service, another set of author information may need to be provided. The lack of standardisation was a problem during the development of this project. However, this situation is improving as a result of Figshare (and other organisations such as Symplectic (symplectic.co.uk) recently adding support for ORCID (orcid.org) researcher IDs [1, 4], which are considered a more standardised way of identifying and attributing authors to research outputs. In the near future, it is hoped that support for authenticating via ORCID and using an ORCID ID when publishing via the Figshare API (instead of the web interface) will also be added.

Some research involves proprietary and/or private data which cannot be shared, but at the same time digital curation is important for the funders of the research. PyRDM is capable of publishing to private Figshare repositories. However, Figshare currently offers limited free private storage space (1 GB) for individual users which it is not generally large enough to store complete modern day simulations, for example. Moreover, the number of collaborators who can view/modify the private data is also limited. Figshare for Institutions, which allows members of an institution to store software and data privately in the cloud, may be a more suitable platform for larger-scale research data management. Its recent adoption by UK-based institutions such as Loughborough University [2] will hopefully further encourage institutions worldwide to integrate a more sustainable research data management framework that can cope with both public and private research outputs and a greater demand for collaboration.

References

[1] Figshare. figshare ORCID integration. http://figshare.com/blog, 2014. 2

[2] Figshare. Loughborough University, figshare, Arkivum and Symplectic announce pioneering research data management solution. http://figshare.com/blog, 2014.

[3] C.T. Jacobs, A. Avdis, G.J. Gorman, and M.D. Piggott. PyRDM: A Python-based library for automating the management and online publication of scientific software and data. Journal of Open Research Software, 2(e28), 2014.

[4] Symplectic. Elements integrates with ORCID. http://symplectic.co.uk/elementsupdates, 2014.

Time Series RDM Green Shoots Report

Nick Jones

What we did in one sentence

We modified our web-resource so that now users can upload time series data, and methods, and compare them in a basic manner.

Details of funding provided

We were provided with 20k. This was used to recruit a scientific web developer, Dr Philip Knaute.

Context

Univariate time series are sequences of measurements of a quantity: from a patient's hourly blood pressure measurements to stock prices recorded each millisecond. As such they underpin huge areas of science, technology and medicine. My research group has developed a method that allows new (time series) data and models/methods to be automatically placed in the context of past efforts [1, 2]. We recently developed a website (http://www.comp-engine.org/timeseries/) that allows people to download the largest interdisciplinary collection of time-series data and time-series analysis code that we know of. We also go further by allowing users to identify sets of time series (and sets of time-series analysis methods) in our database that are functionally similar to one another. In our proposal we sought to make a first attempt at extending this capability, to allow scientists, and the broader public, to automatically determine how their own methods and data are related to our extensive collections.

Our basic approach exploits feature-based comparison of time series and methods. We subject time series to thousands of distinct data analysis methods where each method takes a time series and returns a single output number. This yields a comprehensive feature vector that is informative about a wide range of structures in time series [1, 2]. We can also take time-series analysis algorithms and, by looking at their output on a set of time series, correspondingly represent them by a feature vector with one element for each output. By comparing feature vectors and using tools from machine learning we can compare and organize both time series and their analysis methods according to their empirical properties and behaviour. Importantly, many time series in our database are generated from various mathematical models (e.g., sets of ODEs, iterative maps, stochastic processes). This allows us to take empirical data and identify the model-generated data that it is most similar to. By these means we have the ability to automatically investigate the structure of connections between different empirical time-series data and models, as well as between a broad and extensive range of scientific time-series analysis methods. As our database continues to grow we will be able to place each new piece of time-series data (be it real-world or model-generated), and data analysis method, in its natural scientific context.

Activities supported

Our three major objectives were to make it possible for users to:

1. automatically profile their time series
2. automatically profile their time series algorithms
3. use these profiles to place their work in the context of others

We have delivered on these three aspects: you can explore the upload and analysis capabilities here. Users can now (after installing the free MATLAB runtime environment) analyse their time series using our bundle of analysis methods and upload the output (plus raw data) to our site while supplying metadata. Our site does a search and returns a list of closest matching time series which can then be explored in their turn. Users can also perform a similar set of steps for time series data. The website has explanatory text but we also supply a small read-me.

How does this support best practice in RDM?

An issue with data sharing is that the benefits are substantially larger for the community than for the sharer. We have made the first steps to giving users extra reasons to share their data: we make the data easy to find and we make it easy for the sharer to learn about their data. We thus doubly incentivize upload: uploading allows analysis of the data and the discovery of nearest neighbours and uploading allows the work to be automatically found by others.

Next steps

At the moment we rely on the MATLAB runtime environment. This was suitable for this short project but it is a large and time-consuming download that will likely put o many users. We are thus seeking resource to make our site quicker by moving some of our computations server-side. We will likely have to switch our code over from MATLAB to C in order to take advantage of speed.

References

[1] B. D. Fulcher, M. A. Little, N. S. Jones. Highly comparative time-series analysis: The empirical structure of time series and their methods, J. Roy. Soc. Interface 10, 20130048 (2013).

[2] B. D. Fulcher, N. S. Jones. Highly comparative, feature-based time-series classification. IEEE Transactions in Knowledge and Data Engineering 26, 3026 (2014).

[3] E. Keogh, S. Kasetty. On the need for time series data mining benchmarks: A survey and empirical demonstration. Data Min. Knowl. Disc. 7, 349 (2003).