Integrating Research Data Management into Geographical Information Systems

Christian T. Jacobs, Alexandros Avdis,
Simon L. Mouradian, and Matthew D. Piggott

Department of Earth Science and Engineering, South Kensington Campus,
Imperial College London, London SW7 2AZ, United Kingdom
{c.jacobs10,a.avdis,simon.mouradian06,m.d.piggott}@imperial.ac.uk
http://www.imperial.ac.uk/engineering/departments/earth-science

Abstract. Ocean modelling requires the production of high-fidelity computational meshes upon which to solve the equations of motion. The production of such meshes by hand is often infeasible, considering the complexity of the bathymetry and coastlines. The use of Geographical Information Systems (GIS) is therefore a key component to discretising the region of interest and producing a mesh appropriate to resolve the dynamics. However, all data associated with the production of a mesh must be provided in order to contribute to the overall recomputability of the subsequent simulation. This work presents the integration of research data management in QMesh, a tool for generating meshes using GIS. The tool uses the PyRDM library to provide a quick and easy way for scientists to publish meshes, and all data required to regenerate them, to persistent online repositories. These repositories are assigned unique identifiers to enable proper citation of the meshes in journal articles.

Keywords: Geographical Information Systems, Research Data Management, Digital Curation, Reproducibility, Digital Object Identifier, Online Repositories

1 Introduction

Computer simulations of ocean dynamics are becoming ever more important to predict the effects of global-scale hazards such as tsunamis [13], the influence of marine renewable energy turbines on sediment transport [20], and the dispersal range of nuclear contaminants [6], to name just a few applications. The underlying numerical model behind such simulations often requires a mesh upon which the equations describing the flow dynamics are solved, thereby transitioning from a continuous description of the region of interest (also known as the domain) to a discrete one. An example focussing on the area around the Orkney and Shetland Isles is shown in Figure 1. A mesh for ocean simulations must be of high enough quality to resolve the intricate coastlines and bathymetry [12]. However, creating such a mesh manually is infeasible for large-scale, high-resolution simulations.

Geographical Information Systems (GIS) offer an effective way of processing bathymetry and coastline data to create a geometry with which to work [19].

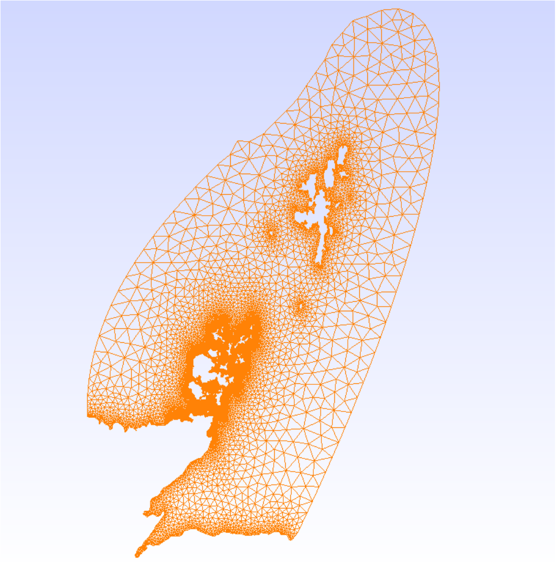


Fig. 1. An example of an unstructured computational mesh which discretises the marine area around the North-East coast of Scotland. The resolution is highest around the Scottish coastline and around the Orkney and Shetland Isles.

A method of producing a computational mesh from this geometry is then required to perform a simulation on it. QMesh [2] is a software package currently being developed at Imperial College London for this purpose. QMesh reads in a geometry defined in the QGIS Geographical Information System software [21], and then converts the geometry into a readable format for the Gmsh mesh generation software [11], which in turn generates the mesh to provide a discrete representation of the domain. Ocean simulations may then be performed with a computational fluid dynamics package.

Publications that are dependant on numerical simulation often provide details of the simulation setups to improve reproducibility and indeed recomputability. However, while a description of the domain may also be given, the mesh that discretises this domain is rarely provided as a supplementary material. This lack of data availability has also been highlighted in many other areas of science [27], [1], [26]. Furthermore, citations to the software used to produce the mesh typically only refer to a generic user manual and contain no information about which version was used. For the purpose of recomputability and reproducibility, it is crucial that researchers provide *all* the data files, as well as the precise version of the software's source code used to produce the output in the first place [8], [5]. In the case of this work, the input data is the geographical information defining the domain, the output data is the computational mesh, and the software is QMesh (and its dependencies).

Despite the need for a more open research environment where software and datasets are shared freely, the level of motivation amongst researchers to do this is generally quite low. This is in part due to the extra effort and time required to gather and publish the data [18], whilst typically gaining little from the process. To encourage the sharing of data and improve its reproducibility and recomputability, it is therefore important to make the publication process more straight forward and swift. This can be effected by the development of research data management tools that readily capture the datasets involved and information about the software being used [25], [18].

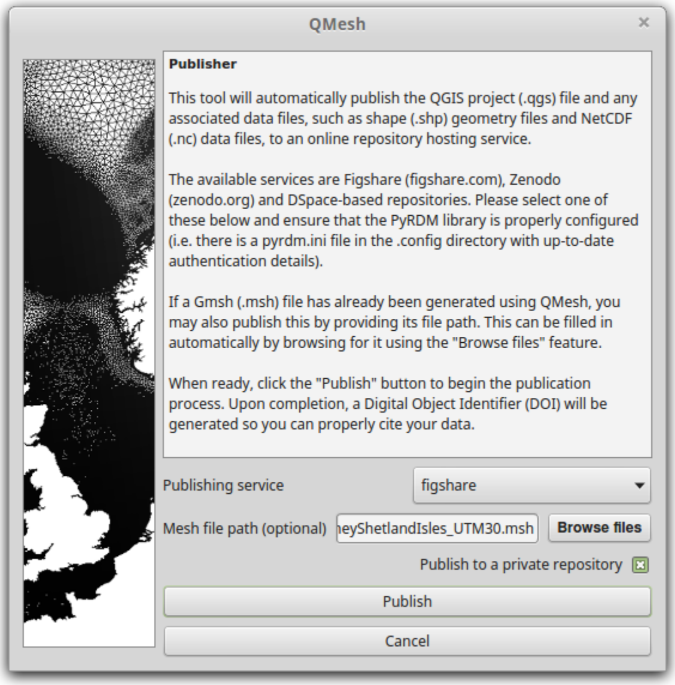
This paper describes the integration of a research data management tool, which uses the PyRDM library [14], into the QMesh software. The tool automates the publication of the QMesh source code, as well as the input and output data for a specified QGIS project, to online, citable and persistent repositories such as those provided by Figshare (figshare.com), Zenodo (zenodo.org) and DSpace-based (dspace.org) hosting services. The tool has both a command line and a graphical user interface, and allows users to publish the software and data at the 'push of a button', thereby facilitating sharing and a more open research environment. In contrast to other software tools that also facilitate the publication of code and datasets, such as Fidgit [24], rfigshare [4], and dvn [17], the QMesh publishing tool incorporates application-specific knowledge to provide a greater amount of automation. For example, the tool is able to parse QGIS project files to automatically determine the relevant input data to publish, rather than the user having to specify the data files manually. Furthermore, this work represents a novel application of research data management and curation software within a GIS environment.

Section 2 describes in greater detail the extensions made to the QMesh software to automate the publication process for the software itself, the input files (for a given QGIS project) and any output files (i.e. the computational mesh). Section 3 presents a realistic example of a scientific workflow involving production of a mesh of a UK coastal region. The data files are read in to QGIS and a mesh is produced. Both the QGIS data and mesh are subsequently published to an online repository provided by Figshare, and a DOI is assigned which can be used to properly cite the data in journal articles. Finally, some concluding remarks are made in Section 4.

2 Integration with QMesh

QMesh features a command line interface (CLI), as well as a graphical user interface (GUI) via a QGIS plugin through which users can select relevant geometry objects and produce a mesh. The integration of research data management techniques into QMesh was achieved by adding a PyRDM-based publishing tool to both of these interfaces.

The tool provides the option of publishing the QMesh software source code and data required to reproduce the mesh to separate online repositories. Users are presented with a simple interface and only have to provide a minimal amount of information; this is illustrated in Figure 2. The publication process itself is handled by the PyRDM library [14] which communicates with an online repository hosting service via its Application Programming Interface (API). The publication process results in a Digital Object Identifier (DOI) [7] being assigned to the repository, with which users can properly cite their research outputs.



Publisher

This tool will automatically publish the QGIS project (.qgs) file and any associated data files, such as shape (.shp) geometry files and NetCDF (.nc) data files, to an online repository hosting service.

The available services are Figshare (figshare.com), Zenodo (zenodo.org) and DSpace-based repositories. Please select one of these below and ensure that the PyRDM library is properly configured (i.e. there is a pyrdm.ini file in the .config directory with up-to-date authentication details).

If a Gmsh (.msh) file has already been generated using QMesh, you may also publish this by providing its file path. This can be filled in automatically by browsing for it using the "Browse files" feature.

When ready, click the "Publish" button to begin the publication process. Upon completion, a Digital Object Identifier (DOI) will be generated so you can properly cite your data.

Publishing service: figshare

Mesh file path (optional): 1eyShetlandIsles_UTM30.msh

Publish to a private repository ☒

Fig. 2. The QMesh publisher tool, which is part of the QMesh QGIS plugin. Users choose the online repository service that they wish to use; by default this is set to Figshare. In addition to the input data files associated with the QGIS project, users may also publish the output data file (i.e. the resulting computational mesh) produced by QMesh, if they so desire. By default, the publication is made public unless the user decides otherwise; in the case of private publication, a DOI is still assigned to the repository, but will not be made active/'live' until the repository is made public.

The publication of data is handled separately to the publication of the QMesh software. In the former case, when a suitable mesh has been produced and is ready to be published, users simply have to provide the QMesh publishing tool with the location of the QGIS project file on the computer's file system when using the CLI. When using the GUI, this location is provided automatically when the project is opened in QGIS. The tool then searches for the `<datasource>` tags in the XML-based project file to determine the location of all the files that the project comprises; these may include shape files that define various layers in the geometry, data files in NetCDF format [23] which define the bathymetry of the ocean, and a multitude of other data formats. Optionally, the location of the Gmsh mesh file may also be provided, thereby publishing the resultant output data along with the files required to produce it. The locations of all these data files, including the QGIS project file itself, are then provided to PyRDM which automatically creates a repository on the hosting service and uploads the files via the service's API. The service then returns a publication ID and a DOI, which is presented to the user for citation purposes. This process is illustrated in Figure 3.

The publication of software involves a similar process, but can currently only be accomplished via the CLI. The user only has to provide the QMesh publishing tool with the location of the software's source code on the computer's file system. The PyRDM library then handles the rest; it determines the exact version of QMesh currently in use using the Git version control system (git-scm.com) [22], and then checks to see whether that version has been published already[1]. If it has, PyRDM retrieves the existing DOI for re-use. If it has not, then PyRDM publishes the source code in a similar fashion to the case of publishing data, as shown in Figure 3. Note that publications in journals would need to reference both the software repository's DOI and the data repository's DOI. There is currently no explicit link that is made between the software and data repositories, unless specified manually.

As demonstrated by Figure 3, the QMesh publishing tool requires minimal user interaction and is largely automated by the PyRDM library. This is important for encouraging the sharing of software and data files, in order to achieve a more open research environment.

3 Workflow Example

To demonstrate an example of a scientific workflow involving mesh generation using GIS, the Orkney and Shetland Isles considered in [2] and [3] are used. The researcher first has to describe the geography of the domain in QGIS and then decide on the area they wish to create a mesh for. The QGIS project for the Orkney and Shetland Isles comprises a number of geometrical layers which define the coastlines (and potentially coastal engineering structures such as marine power turbines), in addition to a NetCDF file which defines the bathymetry of the ocean floor, and another NetCDF file which defines the desired resolution throughout the mesh. These files are shown in Figure 4 beside the area that will be meshed.

The mesh that QMesh produces for this domain (shown in Figure 1) is then used by the researcher in their marine simulations. Once the researcher is ready to publish their results, they upload the data files associated with the production

[1] Repository searching is only available when using the Figshare repository service, due to API limitations explained later in Section 4. PyRDM will publish the software regardless of whether it has been published before when Zenodo or a DSpace-based service is chosen.

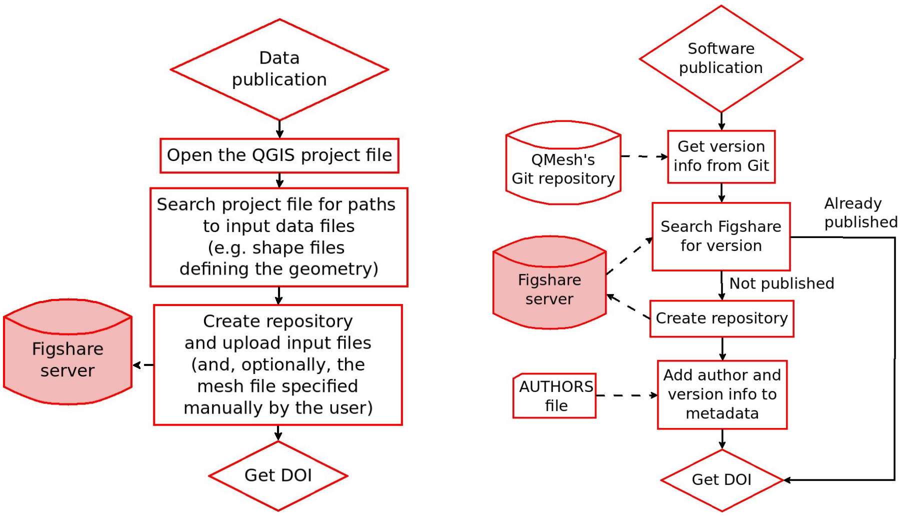
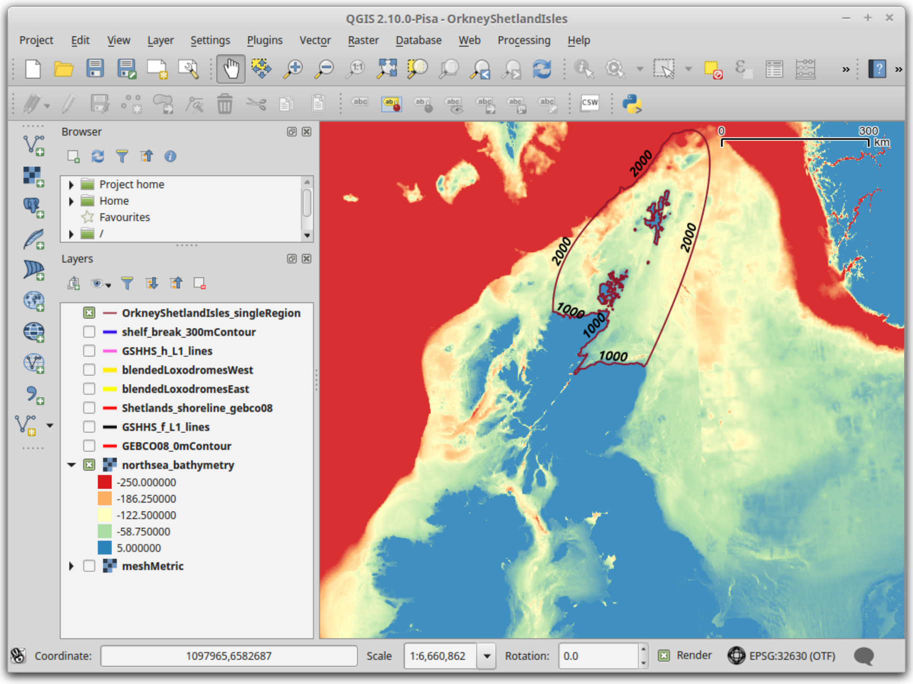

Data publication
Open the QGIS project file
Search project file for paths to input data files (e.g. shape files defining the geometry)
Figshare server
Create repository and upload input files (and, optionally, the mesh file specified manually by the user)
Get DOI
Software publication
QMesh's Git repository
Get version info from Git
Search Figshare for version
Already published
Figshare server
Not published
Create repository
AUTHORS file
Add author and version info to metadata
Get DOI

Fig. 3. The processes behind publishing the QGIS data files (left) and QMesh software source code (right) to Figshare.

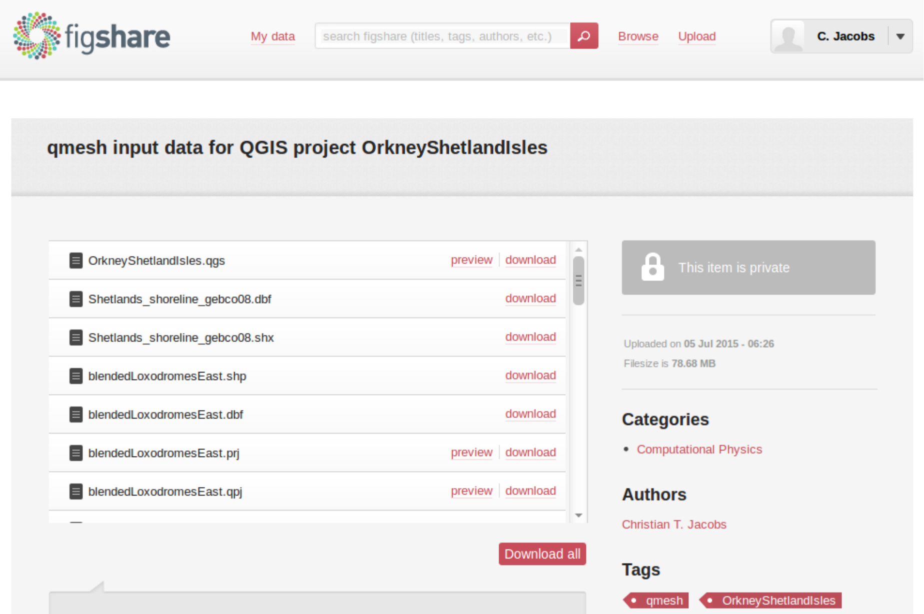


☒ OrkneyShetlandIsles_singleRegion
☐ shelf_break_300mContour
☐ GSHHS_h_L1_lines
☐ blendedLoxodromesWest
☐ blendedLoxodromesEast
☐ Shetlands_shoreline_gebco08
☐ GSHHS_f_L1_lines
☐ GEBCO08_0mContour
☒ northsea_bathymetry
🟥 -250.000000
🟧 -186.250000
🟨 -122.500000
🟩 -58.750000
🟦 5.000000
☐ meshMetric

Coordinate: 1097965,6582687 Scale 1:6,660,862 Rotation: 0.0 ☒ Render EPSG:32630 (OTF)

Fig. 4. Screenshot of the UK region visualised in QGIS. The solid dark purple line defines the area that will be meshed (in this case it contains the Orkney and Shetland Isles). The different files that make up the layers of the geometry are specified in the column on the left-hand side.

of the simulation's mesh to an online repository using the QMesh publishing tool shown in Figure 2 (the CLI may also be used instead of the graphical interface). In this example, it uploads all the files previously mentioned to Figshare. Once uploaded, the files can be downloaded from the Figshare website (see Figure 5) and a DOI is presented to the researcher to share with colleagues and for use in journal publications (see Figure 6).



qmesh input data for QGIS project OrkneyShetlandIsles

OrkneyShetlandIsles.qgs preview | download
Shetlands_shoreline_gebco08.dbf download
Shetlands_shoreline_gebco08.shx download
blendedLoxodromesEast.shp download
blendedLoxodromesEast.dbf download
blendedLoxodromesEast.prj preview | download
blendedLoxodromesEast.qpj preview | download

Download all

This item is private

Uploaded on 05 Jul 2015 - 06:26
Filesize is 78.68 MB

Categories

- Computational Physics

Authors

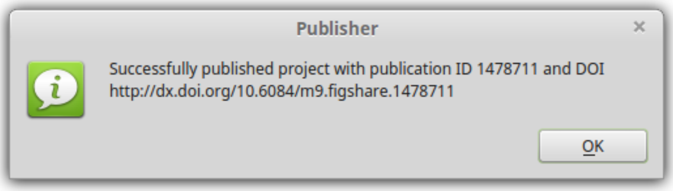
Christian T. Jacobs

Tags

qmesh OrkneyShetlandIsles

Fig. 5. A screenshot of the resulting repository on the Figshare website, with the files readily available to download. The QMesh publishing tool automatically assigns a title and tags to the repository based on the QGIS project's name.

The version of the QMesh software's source code that is used should also be published, in a separate repository to the data. However, it should be noted that publishing the QMesh source code may not be enough to reproduce the exact same mesh without also knowing the versions of its dependencies. For example, different versions of Gmsh may produce slightly different meshes as a result of algorithmic improvements within the software. It is therefore important that such information be recorded in some way to further improve the degree of reproducibility. For example, ideally Gmsh would also have a similar system for publishing the current version of its source code in use.



Successfully published project with publication ID 1478711 and DOI http://dx.doi.org/10.6084/m9.figshare.1478711

Fig. 6. A Figshare publication ID and a DOI are assigned to each repository, and presented to the researcher once the publication process is complete.

4 Discussion and Conclusions

Throughout the production of the PyRDM-based publishing tool for QMesh, several issues were encountered which largely stemmed from a lack of standardisation and support in the repository hosting services' APIs. For example, in order for PyRDM to attribute authors to the software repository on Figshare, all authors of QMesh must provide their Figshare author IDs in the `AUTHORS` file that is part of the QMesh source code. Unfortunately, another different set of author IDs would need to be provided when using a different repository service such as Zenodo, which is inconvenient and requires all authors of QMesh to have accounts across all the supported services. A more standardised way of identifying and attributing authors to research software and data would be to use ORCID (orcid.org) researcher IDs. Figshare has recently added support for authenticating with ORCID IDs via its web interface [9], and it is hoped that ORCID authentication via the Figshare API will also be added for the benefit of PyRDM. Another example, this time involving lack of API support, is the current inability to search for an existing repository with the Zenodo API. Further developments are necessary in this area to enrich the publication process and improve automation.

The production of meshes can involve proprietary and/or private data which cannot be published openly, but at the same time sharing all research output is becoming a common requirement imposed by research funders. The QMesh publishing tool comes with the option of publishing the data to private repositories. However, with some services the private storage space is rather limited, and typically not large enough to store high quality mesh files for realistic ocean simulations. For example, the free private storage space offered by Figshare is 1 GB at the time of writing this paper, with a 250 MB individual file size limit[2]. Furthermore, only a maximum of 5 collaborators can be given access to a private repository. In contrast, the integration of Figshare for Institutions [10] offers a more suitable platform for larger-scale research data management. This project enables researchers at an institution to publish to private repositories hosted in the cloud. This is considerably more sustainable for GIS projects and mesh

[2] `http://figshare.com/pricing`

generation that can involve very large file sizes, both public and private data, and collaboration amongst many researchers and research groups.

In conclusion, the integration of a publishing tool in a Geographical Information System has helped to mitigate one of the reasons why researchers tend not to publish their software and data; that is, it is time-consuming to do so with little reward. The new QMesh publishing tool makes publishing a computational mesh and associated data files easy and largely effortless through the addition of a significant amount of automation. Furthermore, the use of online repository services enable more formal citation of all research outputs through the use of DOIs. However, it is the responsibility of the scientific community to encourage and provide incentives for the openness and public availability of this software and data, in order to overcome the barrier of lack of motivation to publish.

Acknowledgments CTJ was funded by an internal grant entitled "Research data management: Where software meets data" from the Research Office at Imperial College London. Part of the work presented in this paper is based on work first presented in poster form at the International Digital Curation Conference (IDCC) in February 2015 [16], and in a PyRDM project report [15]. The authors would like to thank the two anonymous reviewers of this paper for their feedback.

References

1. Alsheikh-Ali, A.A., Qureshi, W., Al-Mallah, M.H., Ioannidis, J.P.A.: Public Availability of Published Research Data in High-Impact Journals. PLoS ONE 6(9), e24357 (2011)
2. Avdis, A., Hill, J., Jacobs, C.T., Kramer, S.C., Candy, A.S., Gorman, G.J., Piggott, M.D.: Efficient unstructured mesh generation for renewable tidal energy using Geographical Information Systems (In Preparation)
3. Avdis, A., Jacobs, C.T., Hill, J., Piggott, M.D., Gorman, G.J.: Shoreline and Bathymetry Approximation in Mesh Generation for Tidal Renewable Simulations. In: Proceedings of the 11th European Wave and Tidal Energy Conference (Accepted)
4. Boettiger, C., Chamberlain, S., Ram, K., Hart, E.: rfigshare: an R interface to figshare.com. (2014), `http://CRAN.R-project.org/package=rfigshare`, r package version 0.3-1
5. Buckheit, J.B., Donoho, D.L.: WaveLab and Reproducible Research. In: Antoniadis, A., Oppenheim, G. (eds.) Wavelets and Statistics, Lecture Notes in Statistics, vol. 103, pp. 55–81. Springer, New York (1995)
6. Choi, Y., Kida, S., Takahashi, K.: The impact of oceanic circulation and phase transfer on the dispersion of radionuclides released from the Fukushima Dai-ichi Nuclear Power Plant. Biogeosciences 10, 4911–4925 (2013)
7. Davidson, L.A., Douglas, K.: Digital Object Identifiers: Promise and Problems for Scholarly Publishing. Journal of Electronic Publishing 4(2) (1998)
8. de Leeuw, J.: Reproducible Research: the Bottom Line. Department of Statistics Papers, University of California (2001), `http://escholarship.org/uc/item/9050x4r4`

9. Figshare: figshare ORCID integration. Figshare blog, http://figshare.com/blog (2013)
10. Figshare: Loughborough University, figshare, Arkivum and Symplectic announce pioneering research data management solution. Figshare blog, http://figshare.com/blog (2014)
11. Geuzaine, C., Remacle, J.F.: Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering 79(11), 1309–1331 (2009)
12. Gorman, G.J., Piggott, M.D., Wells, M.R., Pain, C.C., Allison, P.A.: A systematic approach to unstructured mesh generation for ocean modelling using GMT and Terreno. Computers & Geosciences 34(12), 1721–1731 (2008)
13. Hill, J., Collins, G.S., Avdis, A., Kramer, S.C., Piggott, M.D.: How does multiscale modelling and inclusion of realistic palaeobathymetry affect numerical simulation of the Storegga Slide tsunami. Ocean Modelling 83, 11–25 (2014)
14. Jacobs, C.T., Avdis, A., Gorman, G.J., Piggott, M.D.: PyRDM: A Python-based library for automating the management and online publication of scientific software and data. Journal of Open Research Software 2(1), e28 (2014)
15. Jacobs, C.T., Avdis, A., Gorman, G.J., Piggott, M.D.: RDM Green Shoots Project Report: Research data management: Where software meets data (2014), http://dx.doi.org/10.6084/m9.figshare.1269127
16. Jacobs, C.T., Avdis, A., Gorman, G.J., Piggott, M.D.: PyRDM: A library to facilitate the automated publication of software and data in computational science. Poster presentation at the 10th International Digital Curation Conference (2015), http://dx.doi.org/10.6084/m9.figshare.1318710
17. Leeper, T.J.: Archiving Reproducible Research with R and Dataverse. The R Journal 6(1) (2014), http://journal.r-project.org/archive/2014-1/leeper.pdf
18. LeVeque, R.J., Mitchell, I.M., Stodden, V.: Reproducible Research for Scientific Computing: Tools and Strategies for Changing the Culture. Computing in Science & Engineering 14(4), 13–17 (2012)
19. Li, R.: Data Models for Marine and Coastal Geographic Information Systems, chap. 3. CRC Press (2000)
20. Martin-Short, R., Hill, J., Kramer, S.C., Avdis, A., Allison, P.A., Piggott, M.D.: Tidal resource extraction in the Pentland Firth, UK: potential impacts on flow regime and sediment transport in the Inner Sound of Stroma. Renewable Energy 76, 596–607 (2015)
21. QGIS Development Team: QGIS Geographic Information System. Open Source Geospatial Foundation (2009), http://qgis.osgeo.org
22. Ram, K.: Git can facilitate greater reproducibility and increased transparency in science. Source Code for Biology and Medicine 8(7) (2013)
23. Rew, R.K., Davis, G.P.: NetCDF: an interface for scientific data access. IEEE Computer Graphics and Applications 10(4), 76–82 (1990)
24. Smith, A.: Fidgit - DOIs for code. figshare (2013), http://dx.doi.org/10.6084/m9.figshare.828487
25. Stodden, V., Bailey, D., Borwein, J., LeVeque, R.J., Rider, W., Stein, W.: Setting the Default to Reproducible: Reproducibility in Computational and Experimental Mathematics. Tech. rep., Institute for Computational and Experimental Research in Mathematics (ICERM) (2013), http://www.davidhbailey.com/dhbpapers/icerm-report.pdf
26. Vines, T.H., Andrew, R.L., Bock, D.G., Franklin, M.T., Gilbert, K.J., Kane, N.C., Moore, J.S., Moyers, B.T., Renaut, S., Rennison, D.J., Veen, T., Yeaman, S.: Man-

dated data archiving greatly improves access to research data. The FASEB Journal 27(4), 1304–1308 (2013)

27. Whitlock, M.C., McPeek, M.A., Rausher, M.D., Rieseberg, L., Moore, A.J.: Data Archiving. The American Naturalist 175(2), 145–146 (2010)