Recognising the Importance of Software in Research – Research Software Engineers (RSEs), a UK Example

Open Science Monitor Case Study

Jon Switters, David Osimo
January – 2019

Research and Innovation

Recognising the Importance of Software in Research – Research Software Engineers (RSEs), a UK Example - Open Science Monitor Case Study

PDF	ISBN 978-92-76-00896-5	Doi: 10.2777/787013	KI-01-19-256-EN-N

Recognising the Importance of Software in Research – Research Software Engineers (RSEs), a UK Example

Open Science Monitor Case Study

Table of Contents

ACKNOWLEDGEMENTS

The case study part of Open Science Monitor led by the Lisbon Council together with CWTS, ESADE and Elsevier.

Authors

Jon Switters – The Lisbon Council

David Osimo – The Lisbon Council

1.   ABOUT RESEARCH SOFTWARE ENGINEERS IN OPEN RESEARCH

The importance of software in research....

Software plays an extremely important role in research. In 2014, the Software Sustainability Institute (SSI), a UK-based organisation dedicated to cultivating and improving research software to support world class research[1], carried out a survey of 15 Russell Group universities[2]. The survey found that 92% of researchers use software in the research activities. Furthermore, **7 out of 10 UK researchers indicated that it would be impossible to conduct their research without it**, with 56% of the respondents stating that they developed their own software.

Although the importance of software to research is clear, **the skills needed to undertake this role and the career pathways in this field lack formality and do not receive the recognition they deserve** in relation to their contribution to research. According to the results of the nationwide survey of 335 Research Software Engineers (RSEs) carried out by the Software Sustainability Institute in January 2016, 88% stated that they had contributed to research that had been published, however, 24% of these RSEs were not acknowledged for their work.[3]

Quality software as a bridge to open research....

The strong link between the role of software experts and open research should be highlighted. Software helps to document a research process and can therefore help others to reproduce what has been developed (reusability). A lack of software sharing can create a barrier to reusability. Furthermore, by making the methods, models and analysis open to others (accessibility), further progress can be made in gathering knowledge and making new discoveries, an approach that is in line with the key principals of the FAIR (Findability, Accessibility, Interoperability and Reusability) for scientific data management and stewardship and open science[4].

Current situation for software experts in research....

Whilst software experts are employed on specific research projects, they also tend to experience poor conditions of employment in comparison to other research roles. Any researcher wanting to employ a research software engineer for a project has to overcome a series of hurdles including restrictive funding policies and **a general culture in universities that fails to recognise the importance of software to research**. However, researchers often lack the appropriate skills to develop software properly or to write specifications for the software experts. They are also unlikely to be aware of the most recent software available that could benefit their work. Postdocs are often the port of call for lead researchers who seek assistance with research software as they are often assessed on the amount of research papers that they write rather than the quality of their code. This means that the **software developers are often locked into a career with no clear**

---

[1] The website of the UK Software Sustainability Institute: https://www.software.ac.uk/about
[2] Russel Group universities is a selection of 24 leading UK universities committed to high levels of research. For more information about the group see https://russellgroup.ac.uk/
[3] Philippe, O., Chue Hong, N. & Hettrick, S. Preliminary analysis of a survey of UK Research Software Engineers. 2016. 4th Workshop on Sustainable Software for Science: Practice and Experience
[4] Wilkinson, M.; Dumontier, M.; Aalbersberg, IJsbrand J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.W.; da Silva Santos, L.B.; Bourne, P.E.; Bouwman, J.; Brookes, A.J.; Clark, T.; Crosas, M.; Dillo, I.; Dumon, O.; Edmunds, S.; Evelo, C.T.; Finkers, R.; Gonzalez-Beltran, A.; Gray, A.J.G.; Groth, P.; Goble, C.; Grethe, J.S.; Heringa, J.; 't Hoen, P.A.C; Hooft, R., Kuhn, T.; Kok, R.; Kok, J.; Lusher, S.J.; Martone, M.E.; Mons, A.; Packer, A.L.; Persson, B.; Rocca-Serra, P.; Roos, M.; van Schaik, R.; Sansone, S.; Schultes, E.; Sengstag, T.; Slater, T.; Strawn, G.; Swertz, M.A.; Thompson, M.; van der Lei, J.; van Mulligen, E.; Velterop, J.; Waagmeester, A.; Wittenburg, P.; Wolstencroft, K.; Zhao, J.; Mons, B. (15 March 2016).The FAIR Guiding Principles for scientific data management and stewardship

pathway for progression, being judged on research that they themselves are not carrying out, creating a community that is transient and largely unrecognised. As a result of this situation, access to software experts is greatly limited with many researchers also experiencing problems in retaining these individuals in their projects. The lack of appropriate skills can lead to the development of research software that is not reliable and that cannot be re-used easily.[5]

The campaign to support software experts in the UK....

A workshop was held in March 2012 to encourage researchers and developers to start talking and exchanging ideas. During this discussion, the group identified that, in addition to the lack of recognition for software experts in academia, there was also no clear job title. A further study in 2014 identified 200 different job titles in a sample of 400 academic job advertisements related to software development. A need for consensus was obvious.

Following this research, the group agreed on **the title of Research Software Engineer, a title that stresses the two skills that make this role unique: a clear understanding of both research and software engineering.**[6] Whilst the title Research Software Engineer was coined at this stage, the role has been around for many years under a variety of different names, making it difficult for recruiters to advertise positions correctly and complicated for job hunters to find appropriate positions.

The UK Research Software Association (UKRSE)....

A year later, in 2013, a second workshop brought together over 50 people who identified themselves under the job title of Research Software Engineer. Most thought that they were alone in this job role and the workshop gave them an opportunity to exchange experiences and expertise. This community and the efforts of those involved led to the creation of the United Kingdom Research Software Engineers Association (UKRSE) in January 2014, dedicated to raising awareness about RSEs and the important work that they do.[7]

Research Software Groups, a model to support RSEs....

In parallel to the workshops and an intense communication campaign to lobby for more recognition for RSEs, a new model was developed to better organise software experts in academia: Research Software Groups. **Research software groups aggregate RSEs in one place, offering them stable access to different employment opportunities within research projects.** They offer RSEs the chance to develop high-quality code without the distractions of dealing with the career ambitions of the experienced researchers. By doing this, RSEs are made available to research projects that may not have the sufficient need or resources to hire a full-time RSE of their own. The research software groups also cater for those research projects that may need an RSE on an ad hoc basis, e.g. shorter-term positions. The offer of a more long-term career prospect for RSEs is more likely to allow universities to keep good quality candidates. Furthermore, the Research Software Groups can create a culture of sharing and exchange amongst RSEs, leading to better quality code and more reliable research outcomes.

In the initial model established in the UK, the RSEs operate within a university or research organisation and have the following two main functions:

---

[5] Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network

[6] European Commission Expert Group on FAIR Data (November, 2018). Turning Fair into Reality - Final Report and Action Plan from the European Commission Expert Group on FAIR Data. European Commission Directorate General for Research and Innovation

[7] The website of the United Kingdom Research Software Engineers association: https://rse.ac.uk/

1) **Service function** – supporting researchers to develop and maintain research software.

2) **Research function** – working with academics to help apply for grants and produce research publications and outputs.

Figure 1 Research Software Groups: The Mixed Funding Approach
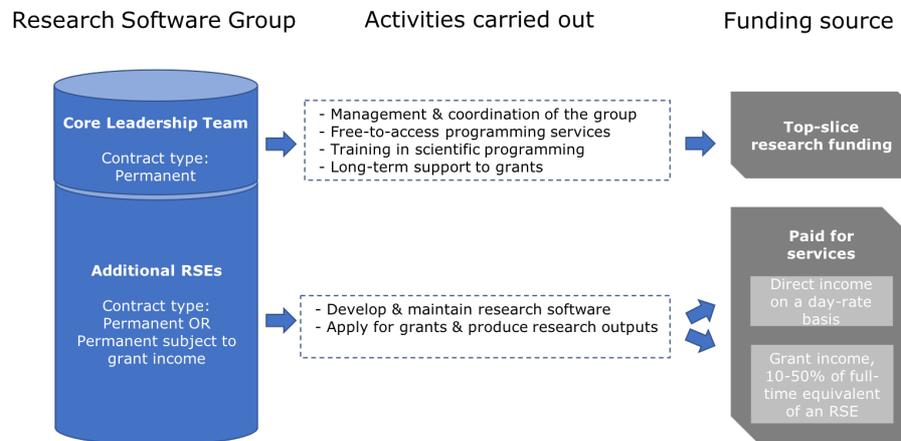
The dual function of RSEs is critical to the success of the initiative and to generating well-designed software that will in turn increase the scope, productivity, reliability, replicability and therefore openness of research. The model can be funded in a variety of different ways. Figure 1 (above) shows an example of the mixed-funding approach, used in some UK-based Research Software Groups.

The mixed funding approach follows a two-tier structure. A core leadership team manages and coordinates the group. A pool of RSEs develops research software, provides training in software engineering and long-term support to grants. They are contracted on a permanent or permanent-subject-to-grant-income basis and are funded through paid-for-services, either through a day-rate or through grant income. This structure allows Research Software Groups to be able to respond to the demand for services and to gradually grow and incorporate more RSEs.[8] The model used to fund Software Research Groups is adaptable and may vary from group to group. Some groups, for example, are entirely self-sufficient such as the group at the University of Southampton.[9]

2. DRIVERS

There are a number of different drivers that led to the campaign to support RSEs and the establishment of research software groups in the UK. Firstly, the research carried out showed that **there was a real problem that needed to be addressed**. There were potentially thousands of people carrying out the role of an RSE, but they had no community to join. They thought that they were on their own and were keen for their fundamental contribution to research to be recognised.

The workshops held in 2012 and 2013 were the triggers to initiate work in this area. They tested the level of interest amongst the community and they put actions in place to work towards supporting RSEs. One of the outcomes of the 2013 workshop was the creation of the **UKRSE Association, an official platform through which the efforts to support RSEs across the UK and beyond could be channelled**.

---

[8] Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network
[9] The website of the Research Software Groups at the University of Southampton: https://rsg.soton.ac.uk/

The advocacy campaign through which the group engaged higher education media outlets, such as speaking at conferences and the creation of regular blog posts and news articles, **also helped to raise the awareness of the role of RSEs and the need to support them**. It also served as a means to reach out to those people who identified as RSEs, but who had not been able to identify one specific job title for the work that they were doing.

Financial support from funding organisations, such as the Engineering and Physical Sciences Research Council (EPSRC), has also been a key driver to push forward different initiatives in this area. These organisations were interested from the start. Despite a large amount of software being used in research projects, reusability is low. Funding organisations often find themselves funding the same software that has already been developed with slight modifications. They saw this as an opportunity to support RSEs and therefore to support the development of quality, open and reusable software. Their involvement also gave the movement more credibility, particularly when organising meetings and attempting to attract interest. An example of interest and support from these bodies can be seen from the EPSRC who initiated a fellowship programme in 2015 which provided funding for a period of five years for a fellow and staff member. A second programme was launched in 2017[10], indicating the success of the initiative.

The involvement of large industry players, such as Google, Microsoft and Amazon, has also helped fund certain activities to support RSEs, particularly the annual event. It is also a sign that there is a general interest (academia, funding bodies, industry) in supporting those people who have a good understanding of both research and software.

Finally, the **democratic and open approach** that was used to set up the UKRSE and the different projects that are carried out through the organisations has been fundamental for it to gain credibility and maintain focus on the key issues that face the RSEs in the UK. The committee members are elected democratically with committee chairs elected for a period of two years. None of the committee members are paid for the work that they carry out and the members of the association are not charged. Listening to the community has proved to be key in supporting the causes that really matter for the target group. In order to reach a more effective model of working, the UKRSE is now carrying out the necessary administrative work to become a charitable organisation in the UK.

3. BARRIERS

The work of the UKRSE and the Research Software Groups is still in its early days and there is still lots of work to be done to continue promoting the role of the RSEs and generating awareness of their importance to ensure high-quality research outputs. Whilst the aforementioned drivers need to be capitalised, there are also a number of barriers that need to be overcome.

One of the main barriers is **the general lack of awareness of the importance of software in research**. A vast majority of researchers are unaware of the role that software plays in supporting their research projects and their reliance on it to generate quality research outputs. Investment in this area is often seen as a waste of money, money that, in the opinion of some, should be focussed towards "traditional" research activities. They fail to see that investment in this area would advance and improve the quality of their research. This is not only a problem in the UK, but across the board in the world of research. Therefore, the link between reliable software and reliable results needs to be further stressed.[11]

---

[10] The website of the Engineering and Physical Research Council (EPSRC): https://epsrc.ukri.org/funding/calls/research-software-engineer-fellowships-ii/

[11] Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network

The heterogenous nature of universities in the UK is also a barrier in expanding the Research Software Group model more quickly and efficiently. The decision-making structure varies from university to university, often with the need to address multiple layers of management across the department, school, faculty and university. Communicating with the right people (e.g. those who have decision-making capability) is therefore difficult. The difference in university structures can also be seen in the way they recruit RSEs. A variety of different employment solutions have been introduced in an attempt to meet the needs of the research projects whilst adhering to restrictions from local finance departments, university culture and funders.

The current incentive system within universities is also proving to be a barrier to the recognition of the work of RSEs. **Researchers are usually judged on the number of papers they write rather than the quality of their code or resultant software**, therefore investment and dedication to the latter is rarely a priority.

Whilst the definition of RSEs becomes ever-increasingly more known and recognised, there needs to be some sort of agreement from funding organisations on how to classify this group of professional with regards to grant justification. For example, in the UK, depending on the university and funding source, RSEs cannot always claim "research overheads" which makes them financially less attractive than standard researcher positions. An agreement needs to be reached (at a central level) so as to avoid confusion on the nature of the role and how it can be funded.

Finally, **academia faces stiff competition from the higher salaries that are offered by industry**, where companies recognise the value of software experts. According to reed.co.uk, in 2018, the average salary for a software developer was £54,089[12] whilst the majority of RSEs working in academia are paid between £30,000-34,000[13]. Universities will have to improve working conditions and improve the recognition of RSEs if they are to compete.
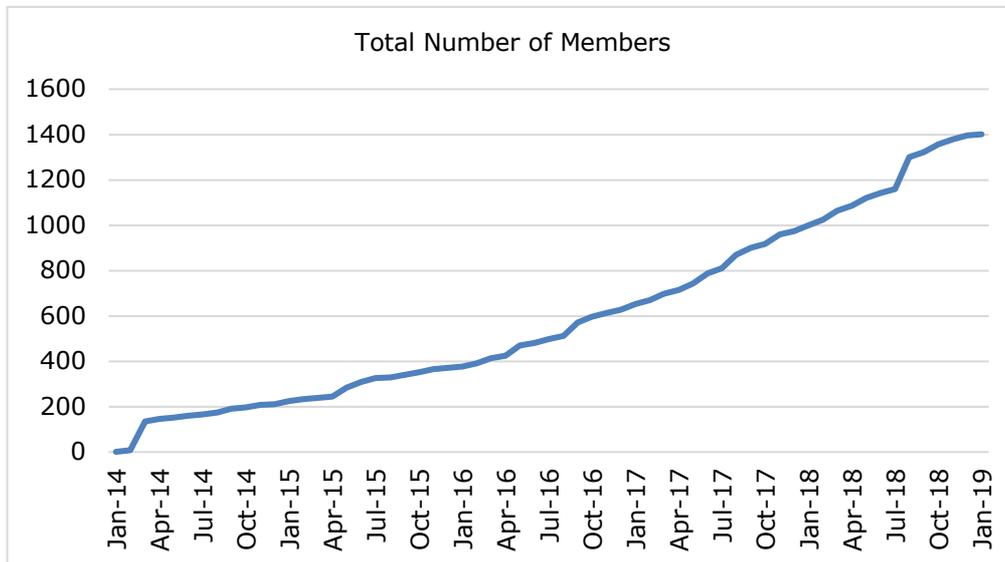
4. IMPACT

In order to evaluate the impact of the initiatives that have been implemented to support RSEs in the UK, the results should be broken down according to the different activities. It should also be highlighted that this movement began fairly recently (2012), therefore it will take time to be able to see the long-term results of such an initiative.

Firstly, the impact and the success of the UKRSE is clear as an organisation that supports RSEs. Figure 2 shows the growth in membership of the UKRSE Association from its creation in 2014 to date.

---

[12] Reed Salary checker: https://www.reed.co.uk/average-salary/it-telecoms. An update (2018 values) of the research carried out in the publication Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network
[13] RSE Survey 2016 data set. https://github.com/softwaresaved/RSE-Survey-2016

Figure 2 Membership of the UKRSE Association

Source: University of Southampton (on behalf of the UKRSE Association)

Within the first week of operation, the *association* gained 50 members. This number has grown steadily over time reaching 1401 in January 2019.[14]

The impact of the association can also be seen through the popularity of the events that organised, namely the yearly UKRSE conference. In 2016 and 2017, the conference sold out two months before the event. In 2018, the capacity of the event was doubled and the event sold out again. Furthermore, the event attracted attendees from 16 different countries and from as far away as Australia, a hint towards the international interest that the association is also generating. The association also includes members from over 20 countries.

Associations have also been created in a number of other countries such as Germany, the Netherlands, Norway and Australia, following the UK model. There is a particularly strong link between the UKRSE and the Workshop on Sustainable Software for Science: Practices and Experiences (WSSSPE)[15], based in the United States, a community-driven organisation that promotes sustainable research software.[16]

The success of the UKRSE has also led to the launch of the Fellowship Programme, with calls in 2015 and 2017. Demand for the original call was high with 211 people applying for the three places advertised. Due to the interest shown by the community, the Engineering and Physical Research Council (EPSRC) who fund the programme increased the funding to cover four RSEs in the second call, providing further employment opportunities within UK academia.

---

[14] Information on membership numbers was provided by the University of Southampton on behalf of the UKRSE Association.

[15] http://wssspe.researchcomputing.org.uk/

[16] Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network

It is also important to look at the **impact and growth of the Research Software Groups that have been established around the UK**. The model (previously described in the case study) was tested at University College London (UCL) in order to create an institution-wide software group. Due to the success of the model at UCL (with three funded posts growing to include an additional 18 roles funded by grants (at time of study)), other universities have created their own groups including Manchester[17], Sheffield[18], Bristol[19], Cambridge, Southampton[20]. In total there are 23 groups across the UK with two thirds of group leaders stating that demand currently outstrips supply. Further growth in the number of these groups in the UK is expected over the coming years.

The potential impact on industry can also be seen by the strong interest shown from some key industry big players such as Google, Microsoft, Intel and Amazon. They have provided support in organizing and speaking at UKRSE events and some industry representatives even form part of the UKRSE Committee. The benefits for industry are quite clear, these firms also actively seek experienced software developers who also understand research. This dual role is essential to help keep them at the forefront of their industry. It also provides an additional career path for software developers who do not wish to continue working in academia.

Finally, the direct results that the Research Software Groups have had at different research organisations across the UK, often improving efficiency of the research projects carried out and, in turn, saving time and money and improving the quality of research outputs. Below, various examples demonstrating the direct impact of RSEs can be seen:

- University of York, UK: "a group of RSEs worked on the industry-sponsored ADDoPT project (https://www.addopt.org/), conducting research to design improved pharmaceutical drugs. They helped to analyse and rework code which modelled van der Waals bonding interactions, **improving the accuracy whilst simultaneously speeding up their simulations** by a factor of more than 4."[21] Quote: Phil Hasnip, The University of York, UK

- Cardiff University, UK: "The 3D black-hole-binary simulation code used by the Gravitational Waves research group at Cardiff University has been further optimised by the Supercomputing Wales RSE embedded within the group, with their work **resulting in code that now runs 50% faster and uses half as much memory**." Quote: Professor Mark Hannam, School of Physics & Astronomy, Cardiff University.

- University of Southampton, UK: "The ForestGrowth SRC algorithm predicts the yield of biomass obtainable under given growth conditions, which is critical knowledge for the development of biofuel crops. We developed the original code, which ran on a single PC, and expanded it to run on the university's high-performance computer (HPC) - **reducing execution time by two orders of magnitude**." Quote: Simon Hettrick and John Robinson.

- EPCC, UK: "The ARCHER eCSE programme funded over 100 RSE projects ranging over a wide range of research areas from climate science, through materials chemistry to computational fluid dynamics. We helped them **improve their HPC software leading to an estimated saving of £24.5M in time** on the UK national HPC service, ARCHER." Quote: Andy Turner

---

[17] The website of the Research Software Groups at the University of Manchester: http://www.itservices.manchester.ac.uk/research/services/software/
[18] The website of the Research Software Groups at the University of Sheffield: https://rse.shef.ac.uk/
[19] The website of the Research Software Groups at the University of Bristol: http://www.bristol.ac.uk/acrc/research-software-engineering/what_is_rse/
[20] The website of the Research Software Groups at the University of Southampton: https://rsg.soton.ac.uk/
[21] Quote from Phil Hasnip, The University of York, UK

- University of Manchester, UK: "The Research Software and Data Science group worked with the Human Brain Project, that conducts research to understand how the brain works. We helped them by leading the team that develops the high-level software for the SpiNNaker neuromorphic computing system. This has resulted in the **design and implementation of high-quality, reliable software** that has resulted in the first brain simulation of an interesting scale to be executed on a neuromorphic system." Quote: Robert Baxter.

Although it is clear that there is still lots more work to do, **there is a strong community growing in the UK to support the RSEs on their recognition, to develop attractive career paths and to establish appropriate reward structures in an attempt to achieve better quality software for better quality research**. This is now the fifth year of the campaign and it is clear that the UK a leading position in the world in recognising and supporting research software engineers[22].

5.   LESSONS LEARNT

The key lessons learnt gathered during the process of promoting the role of the RSEs in the UK include:

- **The importance of the researcher/developer relationship**: in order for software development to be a success in research, a partnership should be built between the researcher and developer. It should not be approached as a one-off transaction.

- **Openness and transparency**: in order to create a self-supporting community, openness and transparency in everything that is done is necessary. This builds trust amongst the people involved in the process and attracts new members. It also ensures that the process does not lose focus of the end goal.

- **Listen to the community**: the importance of listening to the community through open dialogue should not be overlooked. It was by listening to the RSEs that the movement was able to identify the main problems that RSEs face. The fact that the campaign is responding to real problems also makes it more credible and generates trust amongst those who are taking part. It is also important to show stakeholders how their opinions are considered and how actions are adapted according to their input.

- **No "one size fits all" approach to communicating with decision-makers**: reaching the senior-level academics is difficult and not every institution functions in the same way. A tailored communication plan needs to be developed in order to reach these people and to continue lobbying for universities to carry out activities to support software experts.

6.   POLICY CONCLUSIONS

From the broader European policy perspective, stakeholders identified several conclusions. Firstly, **a drastic change in the way researchers are incentivised needs to be implemented**. Other factors should be considered when evaluating the success and career progression of researchers, not merely the amount of papers that they have written: quantity is not always quality.

Funding organisations and bodies at all levels (European, national and regional) should be the first to make a move, changing the model for allocating grants to reflect the important

---

[22] Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). Research Software Engineers: State of the Nation Report 2017. University of Southampton on behalf of the Research Software Engineer Network

role of quality software and the subsequent need to recruit software experts into research projects. **Funding bodies should include RSEs in the preparation and execution of funding calls**. They should also encourage the involvement of RSEs in the evaluation process of calls that have a strong dependence on software development.

If universities were able to cover the full cost of RSEs through grants, it would pressurise them to change their incentive structures and to develop established career paths for the software experts, judging them against metrics that are in line with the work that they carry out. The is a win/win situation for all, it would help universities retain the most skilled staff, provide researchers with the expert software skills that many require in order to achieve quality results, reliable, reusable and more efficient results and provide RSEs with more stable employment opportunities and an attractive career path. **Openness in research needs to be given more importance, otherwise it will be the focus of researchers, simply because they are not evaluated on this.**

Universities should also be encouraged to create more research software groups. They provide a pool of software expertise for many researchers throughout the university. Experience has shown that over time they can also become self-sustaining, although financial support for the staff members may be needed at the beginning to kick-start such an initiative. The exchange of RSEs between different software research groups within the UK and in Europe should be encouraged to increase the community of sharing. A programme such as "ERASMUS for RSEs" could be established to further encourage and incentivise these exchanges.

It is also important to realise that RSEs are not alone. **There are various other roles and groups of professionals that also lack recognition and attractive career paths in academia, such as data stewards and data scientists.** These roles are growing in the scientific community. These roles also need to support and to be taken seriously in modern research communities. Whilst some of the work carried out for RSEs could also be replicated for these types of roles, individual communities will be needed for each.

By gaining recognition for RSEs and establishing an attractive career path for them within academia, it will help to move one-step closer towards a transparent end-to-end research process (idea-publication) where all sources are cited, all versions are logged and everyone involved in the process is recognised, from the researchers themselves to data administrators and technicians.

REFERENCES

European Commission Expert Group on FAIR Data (November, 2018). *Turning Fair into Reality - Final Report and Action Plan from the European Commission Expert Group on FAIR Data.* European Commission Directorate General for Research and Innovation.

Brett A., Croucher M., Haines R., Hettrick S., Hetherington J., Stillwell M., Wyatt C. (2017). *Research Software Engineers: State of the Nation Report 2017.* University of Southampton on behalf of the Research Software Engineer Network

S.J. Hettrick et al (2014). *UK Research Software Survey 2014*

Hettrick, S., Antonioletti, M., Chue Hong, N., Crouch, S., Inupakutika, D., & Parkinson, T. (December 2014). *It's impossible to conduct research without software, say 7 out of 10 UK researchers.*https://www.software.ac.uk/blog/2016-09-12-its-impossible-conduct-research-without-software-say-7-out-10-uk-researchers

Philippe, O., Chue Hong, N. & Hettrick, S. (2016). *Preliminary analysis of a survey of UK Research Software Engineers*. 4th Workshop on Sustainable Software for Science: Practice and Experience

Wilkinson, M.; Dumontier, M.; Aalbersberg, IJsbrand J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.W.; da Silva Santos, L.B.; Bourne, P.E.; Bouwman, J.; Brookes, A.J.; Clark, T.; Crosas, M.; Dillo, I.; Dumon, O.; Edmunds, S.; Evelo, C.T.; Finkers, R.; Gonzalez-Beltran, A.; Gray, A.J.G.; Groth, P.; Goble, C.; Grethe, J.S.; Heringa, J.; 't Hoen, P.A.C; Hooft, R.; Kuhn, T.; Kok, R.; Kok, J.; Lusher, S.J.; Martone, M.E.; Mons, A.; Packer, A.L.; Persson, B.; Rocca-Serra, P.; Roos, M.; van Schaik, R.; Sansone, S.; Schultes, E.; Sengstag, T.; Slater, T.; Strawn, G.; Swertz, M.A.; Thompson, M.; van der Lei, J.; van Mulligen, E.; Velterop, J.; Waagmeester, A.; Wittenburg, P.; Wolstencroft, K.; Zhao, J.; Mons, B. (15 March 2016).*The FAIR Guiding Principles for scientific data management and stewardship*

RSE Survey 2016 data set. https://github.com/softwaresaved/RSE-Survey-2016

Websites consulted during desk research:

https://www.software.ac.uk/about

https://russellgroup.ac.uk/

https://rse.ac.uk/

https://rsg.soton.ac.uk/

http://www.itservices.manchester.ac.uk/research/services/software/

https://rse.shef.ac.uk/

http://www.bristol.ac.uk/acrc/research-software-engineering/what_is_rse/

https://epsrc.ukri.org/funding/calls/research-software-engineer-fellowships-ii/

https://www.reed.co.uk/average-salary/it-telecoms.

http://wssspe.researchcomputing.org.uk/

Software plays an extremely important role in research, a role that is often overlooked within academia. This case study presents the work being carried out in the UK (and beyond) to promote and recognise the role of Research Software Engineers (RSEs) in research, a collective term created to bring together those software experts with an understanding of both research and software engineering. This case study discusses the current challenges faced by RSEs, such as a general lack of recognition of the work they do and a lack of an attractive career path. It also details the drivers behind the movement in the UK, including the creation of the UK Research Software Engineers Association (UKRSE) and the establishment of Research Software Groups within universities, offering RSEs stable access to employment opportunities in research projects and providing researchers access to highly specialised software experts, thus improving the quality of their code and research outputs. Finally, the study covers the key lessons learnt and policy conclusions to further help support RSEs and to contribute to the progress of open science in Europe.

Studies and reports