



# Technological Advances in Remote Collaborations

Rika Kobayashi<sup>1</sup>

Received: 16 August 2021 / Accepted: 23 September 2021 / Published online: 15 October 2021  
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2021

## Abstract

Sustainable scientific software needs a strong collaboration framework to ensure continuity by passing on the tools, skills and knowledge needed to the next generation. The COVID-19 pandemic triggered the unexpected effect of accelerating the development of remote platforms and tools to open up collaborations to a wider global community. In this article we outline the elements needed for such a framework, such as education, tools and community building, and discuss the current advances in technology with a nod to the future.

**Keywords** Remote collaboration · Computational chemistry · Software development · The Future of Meetings · XR

## 1 Introduction

Computational chemistry has been going strong since the 1950s when theoretical chemists first embraced the advent of digital computers to carry out ab initio molecular orbital calculations [1]. The earliest computational chemistry programs, such as POLYATOM [2], ATMOL [3] and Gaussian [4, 5], of which only Gaussian is still in use today, began to appear in the 1970s. Since then, the list of programs has proliferated, including a wider variety of methods, such as molecular mechanics, molecular dynamics and plane-wave codes. Some software packages are still being developed long after the people who started them have gone. Some have fallen by the wayside. This widely recognised problem of continuity in maintaining these packages—keeping up with new operating systems/architectures and support—has been coined software sustainability [6–8]. Software sustainability requires the inextricably linked resources of funding and people. Certainly, in computational chemistry,

---

This article is part of the Topical Collection “New Horizon in Computational Chemistry Software”; edited by Michael Filatov, Cheol. H. Choi and Massimo Olivucci.

---

✉ Rika Kobayashi  
Rika.Kobayashi@anu.edu.au

<sup>1</sup> ANU Supercomputer Facility, Leonard Huxley Building 56, Mills Rd, Acton, ACT 2601, Australia

programming skills do not lead to a secure career path, especially in academia. As Katz [7] recognises, Software Developers need to “keep themselves supported/employed”. In recognition of this, in 2016 the United States government funded the Molecular Sciences Software Institute (MolSSI) [9] to serve “as a nexus for science, education and cooperation serving the worldwide community of computational molecular scientists”. This is the only such initiative in existence to our knowledge. A similar initiative led by Peter Gill was proposed to the Australian Government in 2014 but failed to gain funding. However, recognition and funding are only the beginning. Sustainable scientific software needs a strong collaboration framework.

The 2020 COVID-19 pandemic had an unintended effect of bringing the world closer together. The halt to international travel disrupted scientific exchange and researchers scrambled to find ways of making working virtually more effective and software platforms reinvented themselves. One such endeavour was the Future of Meetings (TFOM) symposium, which was organised by a like-minded community with the aim of investigating best practice in how to work, educate, collaborate and interact virtually. It became a truly cross-disciplinary symposium, encouraging us to explore key themes and share ideas from a variety of people and perspectives we don't normally interact with. The symposium proved highly successful and our findings were made into a report [10] and an invited comment from Nature [11]. One theme of the symposium was “Technology to help us collaborate virtually” and this brought to the fore a range of online tools and initiatives that could provide the infrastructure for scientific software sustainability.

## 2 Education

Probably the foremost issue for the continued development of scientific software is the shortage of programmers. This is recognised worldwide, as reflected in the many articles decrying the shortage of software engineers, e.g. Lee [12, 13] estimates that by the year 2024 that number is expected to reach 1 million. This problem is perceived as being one of education, spawning many government initiatives. In Australia, the government introduced the Coding Across the Curriculum initiative with the aim to promote the teaching of digital technologies, including coding, across the different year levels in Australian schools [14]. In 2016, President Obama proposed a CS For All initiative with a US\$4 billion dollar budget for computer science education in the United States that did not get approved [15]. The European Commission produced a digital education report for Europe in 2019 [16] and a general worldwide overview can be found in a 2019 UNESCO report [17]. Most of these initiatives concern the Tech and IT sector and target languages, such as JavaScript, Java and Python [18]. Few address the shortage of scientific programmers, except in the rapidly growing field of data science. However, when it comes to software in the applied sciences, certainly in high-performance computing (HPC), the majority of programs are written in Fortran and C/C++ [19]. From a survey in 2015, Rouson (Rouson, personal communication) reported on the programming languages used at NERSC (National Energy Research Scientific Computing); Fortran accounted for close to 60%, followed by C++ and C, at about 35% and 31%, respectively. Fortran

was found to be the primary language for 23 of the 36 top codes, yet it is a language that is no longer widely taught. See also the entertaining talk by Roland Lindh in this series [20].

The technical skills shortage has been partly addressed by the rise in online learning [21]—a natural extension of distance education—suited particularly to programming and computer-related courses because of its basis in a digital environment. Worldwide government cuts in funding for education have made distance education more appealing as an easy source of revenue, leading to a highly competitive, and growing, education industry. At the turn of the century, the market for higher education through distance learning was estimated at US\$ 300 billion. In 2019, pre-COVID times, Renub Research estimated that the online education market would reach US\$ 350 billion by 2025 [22]. Renub's report highlighted online course providers Coursera [23] and Udacity [24] and indeed their most popular offerings are computer science related: programming, machine learning/artificial intelligence (ML/AI), data science. The majority of courses, as mentioned above, do not target scientific programmers. To date, the only online Fortran programming courses that could be found online were provided through Udemy [25] and Tutorialspoint [26], though there is further material available online in the form of university lecture notes and tutorial handouts. In fact, many online courses just provide lecture material, often in the form of videos and handouts. The more sophisticated online programming courses leverage the digital environment by embedding interactive exercises, as with the Tutorialspoint Fortran course, assessment and a discussion forum. At this point in time until this skills shortage is redressed, as has been done for the IT sector, passing on these advanced programming skills is in the hands of the scientific community.

### 3 Collaboration Tools

The global COVID-19 lockdowns found many technical platforms and tools coming into being or reinventing themselves to adapt to the remote working landscape as can be seen in Fig. 1 (a composite adapted and updated from several sources [27–29]). Video conferencing platforms became schools, fitness centers, places of worship; social apps, such as for chat and gaming, turned into tools for remote work.

The relevant tools from a programming perspective without a doubt would start with code hosting platforms or version control repositories. These have been in use for many years, starting with early version control systems [30], such as CVS [31] and Subversion [32], which were essentially a mechanism for tracking code revisions. However, as software development expanded and became more complex, often involving many authors working concurrently on different parts of code there was a need to have some form of coordination giving rise to distributed revision control, led by Git from Linus Torvalds in 2005 for development of the Linux kernel [33]. The majority of software packages are probably now hosted on such a code repository, the main ones in the scientific community being Github [34] and Gitlab [35]. This is often coupled with a Continuous Integration system. The idea behind the workflow is to keep a master copy of the code on the

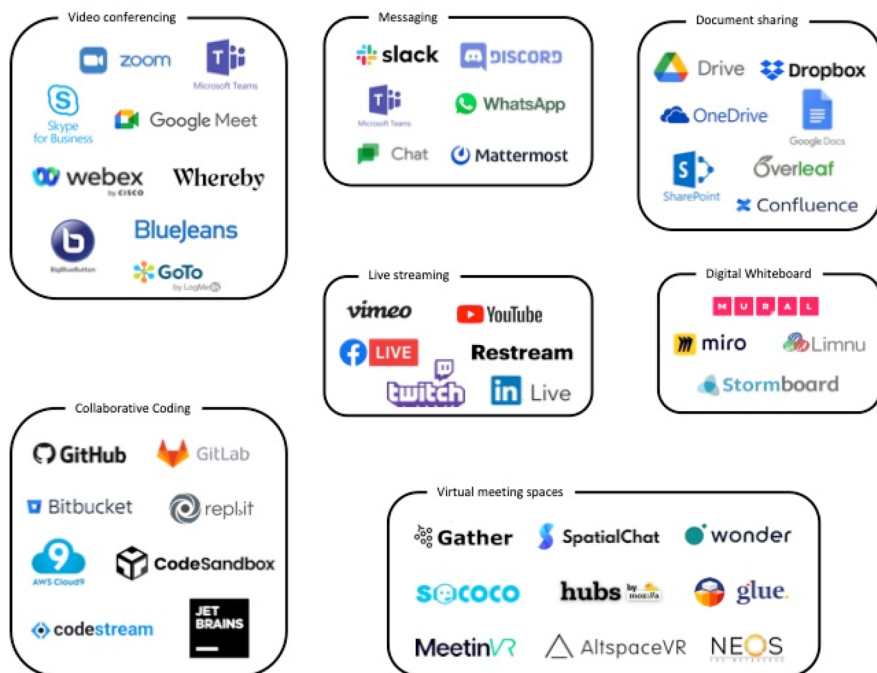


Fig. 1 Remote working tools for collaboration landscape as of 2021

repository which can be downloaded individually and worked on. Changes from the various developers' working copies are then merged back, with the platform providing a mechanism for tracking and resolving dependencies and conflicts. This is the most problematic part of collaborative software development, where independent developers can introduce changes incompatible with each other, e.g. reusing the same variable or changing the underlying structure. One way to mitigate this is to check out the code regularly to keep as close to the master copy as possible, the ideal being to spend less time merging the change than making the change itself—"integration hell". Continuous integration automates this merge to a frequent basis, at least daily, together with running a set of unit tests.

For the extended complicated software suites that make up the bulk of computational chemistry software, new developments can be a long time in the making, rendering "integration hell" unavoidable. Recent times, perhaps in mitigation of this or in conjunction with the popularity of "pair programming", have seen the springing up of real-time collaborative coding platforms where developers work on the same piece of code. This is already something that has been in place for a while with document sharing such as Google Docs [36] and Overleaf [37]. There will probably still be a need for a set of privileged developers to approve code changes but it is believed that the main advantage will be being able to see concurrent development as it happens, making potential conflicts more noticeable. Again, of the most popular collaborative coding tools [38], few target Fortran and

it is still early days so not clear whether this approach can scale to a complicated software suite with a number of modules and developers.

## 4 Community

The biggest concern during the COVID-19 global shutdown was arguably what could be termed the human factor, described variously as lack of connectedness, the need for “real” interactions, and is the main argument for “return to the office” (see next section). There was already a movement towards the formation of global research communities through the rise of collaborative hubs, also known as virtual research environments or science gateways [39]. The idea behind these is to provide a technological infrastructure of shared computational resources: software, data, tools, workflows, HPC access, through a web portal or apps, thus enabling research to a broader scientific community. There now exist many well-established communities, as can be seen in the Special Issue on International Science Gateways 2017 [39], and these have proved successful in some disciplines, notably the Galaxy Project [40], started in 2005, which, according to their latest report [41], had “served hundreds of thousands of users, been used in >5700 scientific publications, and provided 500+ developers with a framework provisioning accessible, transparent and reproducible data analysis”.

There is already some online community through the various code repository platforms but a true collaborative hub should have elements of:

- venue—a central website accessible to all and possibly distributed regional hubs to serve a more localized community;
- repositories for collecting and sharing software, tools, data, workflows;
- active community that can communicate synchronously and asynchronously for the exchange of ideas and training;
- (optionally) access to high-end HPC facilities as part of the workflow.

In the computational chemistry world there are existing initiatives, such as NanoHub [42], and nascent hubs, such as AiiDA [43] and Edison [44]. Nanohub, having been established in 2002, is the oldest of these and describes itself as a science gateway supporting a global Network for Computational Nanotechnology community within a cloud environment. It provides a wealth of resources, including training courses, discussion forums, simulation and modeling tools, and a computing environment in which to run them [45]. Its main usage and success appears, however, to be in delivering courses and enabling simulations—supporting an application community rather than a programming and development one.

On the other hand, AiiDA started from the Quantum Espresso [46] developers community, primarily for building an infrastructure providing tools for designing, deploying and analysing materials science simulations, integrated into HPC environments. However, simultaneous efforts since into education and collaboration have expanded its range into Materials Cloud—a web platform for computational materials scientists to “share their work and promote open science” [47].

Furthermore, the Materials Cloud community have begun taking steps towards defining standards, such as file formats and metadata, to facilitate interoperability. The concept of interoperability, the ability for different groups to exchange data consistently, is recognized as an important part of software sustainability, but as yet has been addressed comparatively little in computational chemistry.

A final quick mention should be given to the talk that initiated and possibly inspired this symposium series “A Web Platform for Scientific Collaborations”, lectured by Cheol Ho Choi [48]. Leveraging the principles of modular environments, they have created a web platform based on sharing and running computational chemistry modules and workflows via a graphical pipeline and opened it up to the wider quantum chemistry community in the hopes of establishing a scientific software ecosystem.

However, these collaboration hubs are still rooted in two-dimensional screens and do not authentically fill the gap of the lamented missing “real” human interactions—the body language, corridor conversations and serendipitous interactions. Continued TFOM activities have allowed us to explore further these social aspects, especially whether extended reality (XR) and immersive technology can help add a social human factor into our virtual offices and conferences. To this end, we have held a variety of events in virtual reality (VR) platforms such as Altspace [49], NEOS [50] and Glue [51]. These were engaging and fully immersive, but the immaturity of the platforms and the technological requirements do not make them practical today. We were able to explore this aspect further by being given the opportunity to discuss “The Future of Meetings: Working in XR?” with the XR Developer Community through a Birds of a Feather Session at SIGGRAPH 2021 [52]. As part of the session, the attendees were polled informally on a variety of questions concerning the state of XR (for the complete set see Ref. [53]). The poll was not a rigorously conducted exercise so not too much can be read into it, but it was indicative that the albeit small sample felt XR was able to substitute “real” human interactions and that the industry believed that we will be seeing meaningful change in the near future (Fig. 2).



**Fig. 2** Informal poll results on various aspects of working in XR from our Birds of a Feather Session at SIGGRAPH 2021

## 5 Virtual Conferences

In the early days of the COVID-19 pandemic an article appeared in Nature headed “A year without conferences?” [54] discussing the impact on researchers and raising the prospect of a need to rethink the concept of meetings. This article was very quickly countered by examples of successful virtual conferences from around the world, spanning many disciplines, notably one from the Virtual Winter School on Computational Chemistry, which has been running annually since 2015 [55]. Traditionally, conferences have been a means for scientific communities to meet and share knowledge, but these stemmed from a time when communication was slow and travel not so easy. However, technological advances as described here have blurred the need to travel to achieve these outcomes. The pros and cons of virtual conferences were explored in depth in the aforementioned Future of Meetings symposium [10, 11] and continue to be explored through various initiatives by the TFOM community [56]. The definite “pro” of virtual conferences has, without doubt, been their accessibility, inclusivity and sustainability. On the whole, virtual attendance figures have been much higher than their in-person equivalents as the cost of travel is no longer a barrier to attendance. Junior researchers have reporting increased confidence and feeling of safety in the virtual environment and there is significantly less harm to the environment. TFOM calculated the symposium produced 1420 kg of CO<sub>2</sub> equivalent compared to an in-person equivalent of 280,000 kg of CO<sub>2</sub>. The biggest disadvantage, certainly for Australia, has been juggling timezones, coupled with the difficulty in separating conference and domestic duties as few international conferences overlap with normal working hours. For developing countries technological accessibility is the biggest problem and a recent a poll of 900 Nature readers [57] cited “poor networking opportunities” as the biggest drawback.

## 6 Future Outlook

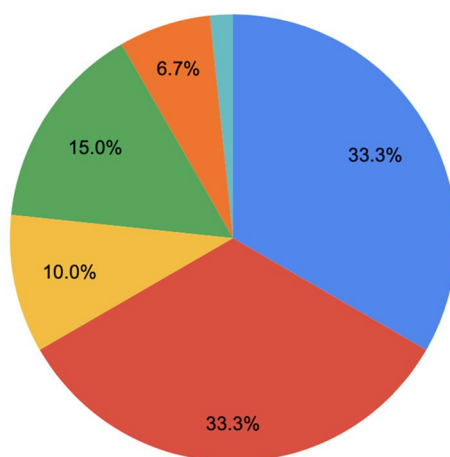
With the effects of the COVID-19 pandemic still being felt around the world today, it is probable that there will be long-lasting changes to the way we work and meet. Major companies, especially the tech giants, are reducing their office space with some going fully remote [58] following the lead in May 2020 of Twitter and Facebook who announced that they would give staff the option to work remotely permanently [59]. Together with the rise in the globally competitive Distance Education industry, online learning is becoming more accepted. The COVID-19 pandemic provided momentum for overcoming the potential barrier to adoption of remote teaching practices that had been considered niche, such as flipped teaching. Such practices have become mainstream, especially as more educators are recognising their effectiveness in this digital age. The perception that online degrees were not “real” degrees is diminishing now that many students have been able to experience direct comparisons. Similarly, Virtual Winter

School, which had been motivated by the desire to make accessible to a wider audience experts in the field they would not normally be able to hear live or interact with, was attended regularly by participants from less advantaged countries. The 2021 School had noticeably more attendance, typically reaching about 200 for most sessions, more than double the usual attendance, and especially from the more established community. The level of engagement demonstrated that the Virtual School is a viable format for fruitful scientific exchange and hopefully this participation level will continue.

There is evidence that scientists want virtual meetings to stay after the COVID pandemic from the Nature poll [57] and reflected in our own survey from TFOM shown in Fig. 3. The concept of “hybrid” is gaining in popularity and it could be that the future will be some mixture of in-person and virtual. However, to be done well, virtual collaborations, whether conferences, meetings or teaching, need more effort, from planning to delivery. TFOM activities have shown how virtual conferences can be effective and subsequently we have been continually called on to advise on best practice for a variety of virtual initiatives. The obvious benefits of accessibility, inclusivity and sustainability, as highlighted in our conference experiences, are still competing with the drawbacks of time-zones, technological accessibility and the human factor, though XR may soon be able to provide a solution for that. TFOM is seeing disheartening signs of people wanting to take what they think is the easy option, i.e. go back to the way things were before. The future will be determined by the people who can see what virtual can do versus the people who see what it can't.

Which scenario would you personally prefer for the future format of conferences? (60 responses)

- Alternating (sometimes physical, sometimes virtual)
- Hybrid ( a mix of online and in-person at the same time)
- Digital first (regardless of physical or virtual)
- Virtual (online)
- Physical (in-person)
- Unsure



**Fig. 3** Attendee preferences for the future format of conferences from The Future of Meetings symposium [10]



## 7 Conclusion

Sustainable scientific software needs a strong collaboration framework. With the lifetime of computational chemistry software packages exceeding that of the developers who began them, there needs to be a mechanism to pass down the generations the software, tools, skills and knowledge to maintain continuity. The COVID-19 pandemic opened up the world to a potentially global community of developers, whether through the imaginative creation of remote collaboration platforms and tools or just by forcing people to take the digital plunge. It is still early days to know what work in the post-pandemic world will look like, whether we just go back to how we used to do things or whether these remote innovations will be embraced and developed further. There is now a wealth of tools out there to help us meet and work, and possibly even develop software, better virtually. Through TFOM, we have demonstrated how these tools can be used to return to a better normal. Now is the time to keep the momentum going and make use of them to build the foundation of a solid and global software development community.

**Acknowledgements** Thanks to the TFOM community, especially Vanessa Moss, Glen Rees and Patrice Rey for accompanying me on this journey, and Aidan Hotan, Chenoa Tremblay, Claire Trenham, Roger Amos and “Earthmark” for cheering us on (and providing invaluable material). Thanks also to Damian Rouson of Berkeley Lab for his excellent “Why Fortran persists” presentation, and the Australian Government Department of Education, Skills and Employment for providing a copy of their Coding Across the Curriculum report.

## Declarations

**Conflict of interest** The author has no conflicts of interest or relevant financial interests to declare.

## References

1. Roothaan CCJ (1951) New developments in molecular orbital theory. *Rev Mod Phys* 23:69. <https://doi.org/10.1103/RevModPhys.23.69>
2. Moskowitz JW, Snyder LC (1977) Polyatom: a general computer program for ab initio calculations. In: Schaefer HF (ed) *Methods of electronic structure theory. Modern theoretical chemistry*, vol 3. Springer, Boston
3. ATMOL. <http://www.chilton-computing.org.uk/acl/applications/qc/p003.htm>. Accessed 23 July 2021
4. Gaussian.com. Expanding the limits of computational chemistry. <https://gaussian.com/>. Accessed 23 July 2021
5. Frisch MJ, Trucks GW, Schlegel HB, Scuseria GE, Robb MA, Cheeseman JR, Scalmani G, Barone V, Petersson GA, Nakatsuji H, Li X, Caricato M, Marenich AV, Bloino J, Janesko BG, Gomperts R, Mennucci B, Hratchian HP, Ortiz JV, Izmaylov AF, Sonnenberg JL, Williams-Young D, Ding F, Lipparini F, Egidi F, Goings J, Peng B, Petrone A, Henderson T, Ranasinghe D, Zakrzewski VG, Gao J, Rega N, Zheng G, Liang W, Hada M, Ehara M, Toyota K, Fukuda R, Hasegawa J, Ishida M, Nakajima T, Honda Y, Kitao O, Nakai H, Vreven T, Throssell K, Montgomery JA Jr, Peralta JE, Ogliaro F, Bearpark MJ, Heyd JJ, Brothers EN, Kudin KN, Staroverov VN, Keith TA, Kobayashi R, Normand J, Raghavachari K, Rendell AP, Burant JC, Iyengar SS, Tomasi J, Cossi M, Millam JM,

- Klene M, Adamo C, Cammi R, Ochterski JW, Martin RL, Morokuma K, Farkas O, Foresman JB, Fox DJ (2016) Gaussian 16, Revision C.01. Gaussian Inc., Wallingford
6. Blanton B, Lenhardt C (2014) A scientist's perspective on sustainable scientific software. *J Open Res Softw* 2:17. <https://doi.org/10.5334/jors.ba>
  7. Katz DS (2018) Fundamentals of software sustainability. <https://danielskatzblog.wordpress.com/2018/09/26/fundamentals-of-software-sustainability/>. Accessed 11 July 2021
  8. Hannay J, MacLeod C, Singer J, Langtangen HP, Pfahl D, Wilson G (2009) How do scientists develop and use scientific software?" In: Proceedings of the 2009 ICSE Workshop on Software Engineering for Computational Science and Engineering, SECSE '09, Washington, DC, USA, IEEE Computer Society, pp 1–8
  9. MolSSI—The Molecular Sciences Software Institute. <https://molssi.org/>. Accessed 23 July 2021
  10. Moss VA, Hotan AW, Kobayashi R, Rees GA, Siegel C, Tremblay CD, Trenham CE, Engelke U, Gray A, Hurley-Walker N, Roos G (2020) The future of meetings: outcomes and recommendations. Zenodo. <https://doi.org/10.5281/zenodo.4345562>
  11. Moss VA, Adcock M, Hotan AW, Kobayashi R, Rees GA, Siégel C, Tremblay CD, Trenham CE (2021) Forging a path to a better normal for conferences and collaboration. *Nat Astron* 5:213–216. <https://doi.org/10.1038/s41550-021-01325-z>
  12. Lee T (2019) How to close the tech skills gap. *Scientific American*. <https://blogs.scientificamerican.com/observations/how-to-close-the-tech-skills-gap/>. Accessed 19 July 2021
  13. Scarpelli B, Miller N, Stephens R (2017) State of the app economy, 5th edn. Act: The App Association
  14. Australian Government Department of Education and Training (2018) Evaluation of the Coding Across the Curriculum program Australia: dandolopartners
  15. Wills B (2016) The United States of coding. *New America*. <https://www.newamerica.org/weekly/united-states-coding/>. Accessed 19 July 2021
  16. European Commission/EACEA/Eurydice (2019) Digital education at school in Europe. Eurydice Report. Publications Office of the European Union, Luxembourg
  17. Storte D, Webb M, Bottino R, Passey D, Kalas I, Bescherer C, Smith J, Angeli C, Katz Y, Micheuz P, Røsvik S, Brinda T, Fluck A, Magenheim J, Anderson B, Fuschek G (2019) Coding, programming and the changing curriculum for computing in schools. Report of UNESCO/IFIP TC3 Meeting at OCCE—Wednesday 27th of June 2018, Linz, Austria
  18. Hughes O (2021) These are the programming languages most in-demand with companies hiring. *TechRepublic*. <https://www.techrepublic.com/article/these-are-the-programming-languages-most-in-demand-with-companies-hiring/>. Accessed 19 July 2021
  19. Partee S (2021) Why are climate models written in programming languages from 1950? PARTEE.IO. <https://partee.io/2021/02/21/climate-model-response/>. Accessed 15 July 2021
  20. Lindh R (2021) Strategies for the OpenMolcas Legacy codes—fifty shades of Fortran [Video]. YouTube. [https://youtu.be/LcDU\\_O4fmXg](https://youtu.be/LcDU_O4fmXg). Accessed 8 Mar 2021
  21. Koksals I (2020) The rise of online learning. *Forbes*. <https://www.forbes.com/sites/ilkerkoksals/2020/05/02/the-rise-of-online-learning/>. Accessed 13 Nov 2020
  22. Renub Research (2019) Online education market & global forecast, by end user, learning mode (self-paced, instructor led), technology, country, company. <https://www.renub.com/online-education-market-p.php>. Accessed 13 Nov 2020
  23. Coursera. <https://www.coursera.org/>. Accessed 31 July 2021
  24. Udacity. <https://www.udacity.com/>. Accessed 31 July 2021
  25. Udemy Fortran Courses. <https://www.udemy.com/topic/fortran/>. Accessed 31 July 2021
  26. Tutorialspoint —Learn Fortran. <https://www.tutorialspoint.com/fortran/index.htm>. Accessed 31 July 2021
  27. Wengi F (2020) Remote work (WFH) tech landscape pauaventures. <https://medium.com/paua-insights/remote-work-wfh-tech-landscape-85627ced3410>. Accessed 23 Oct 2020
  28. Out of office: 65+ startups helping you work from home (2020) CBINSIGHT. <https://www.cbinsights.com/research/remote-work-from-home-market-map/>. Accessed 23 Oct 2020
  29. The ultimate remote work tools landscape. <https://www.holloway.com/g/remote-work/sections/the-ultimate-remote-work-tools-landscape>. Accessed 23 October 2020
  30. Rockkind MJ (1975) The source code control system. *IEEE Trans Softw Eng* 1:364–370
  31. Tichy WF (1985) Rcs — a system for version control. *Softw Pract Exp* 15:637–654
  32. Collins-Sussman B, Fitzpatrick BW, Pilato CM (2004) Version control with subversion, 1st edn. O'Reilly Media, Sebastopol, CA

33. Chacon S, Straub B (2014) Pro git. Apress, Berkeley. <https://doi.org/10.1007/978-1-4842-0076-6>
34. GitHub. <https://github.com/>. Accessed 4 Aug 2021
35. GitLab. <https://gitlab.com/>. Accessed 4 Aug 2021
36. Google Docs. <https://docs.google.com/>. Accessed 4 Aug 2021
37. Overleaf. <https://www.overleaf.com/>. Accessed 4 Aug 2021
38. Jones B (2021) The top 5 online IDEs for in-browser development developer.com. <https://www.developer.com/cloud/top-online-ide/>. Accessed 31 July 2021
39. Dahan M, Pirzl R, Gesing S (2020) International science gateways 2017 special issue. *Futur Gener Comput Syst* 110:320–322. <https://doi.org/10.1016/j.future.2020.04.045>
40. Galaxy Community Hub. <https://galaxyproject.org/>. Accessed 4 Aug 2021
41. Afgan E, Baker D, Batut B, van den Beek M, Bouvier D, Čech M, Chilton J, Clements D, Coraor N, Grüning BA, Guerler A, Hillman-Jackson J, Hiltemann S, Jalili V, Rasche H, Soranzo N, Goecks J, Taylor J, Nekrutenko A, Blankenberg D (2018) The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res* 46:537–544. <https://doi.org/10.1093/nar/gky379>
42. nanoHUB. <https://nanohub.org/>. Accessed 4 Aug 2021
43. Huber SP, Zoupanos S, Uhrin M, Talirz L, Kahle L, Häuselmann R, Gresch D, Müller T, Yakutovich AV, Andersen CW, Ramirez FF, Adorf CS, Gargiulo F, Kumbhar S, Passaro E, Johnston C, Merkys A, Cepellotti A, Mounet N, Marzari N, Kozinsky B, Pizzi G (2020) AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. *Sci Data* 7:300. <https://doi.org/10.1038/s41597-020-00638-4>
44. Edison MQCP-Modular Quantum Chemistry Package <https://mqcp.edison.re.kr/>. Accessed 15 Aug 2021
45. Madhavan K, Zentner L, Farnsworth V, Shivarajapura S, Zentner M, Denny N, Klimeck G (2013) nanoHUB.org: cloud-based services for nanoscale modeling, simulation, and education. *Nanotechnol Rev* 2:107–117. <https://doi.org/10.1515/ntrev-2012-0043>
46. Giannozzi P, Baroni S, Bonini N, Calandra M, Car R, Cavazzoni C, Ceresoli D, Chiarotti GL, Cococcioni M, Dabo I, Dal Corso A, de Gironcoli S, Fabris S, Fratesi G, Gebauer R, Gerstmann U, Gougousis C, Kokalj A, Lazzeri M, Martin-Samos L, Marzari N, Mauri F, Mazzarello R, Paolini S, Pasquarello A, Paulatto L, Sbraccia C, Scandolo S, Sclauzero G, Seitsonen AP, Smogunov A, Umari P, Wentzcovitch RM (2009) QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *J Phys Condens Matter* 21:395502. <https://doi.org/10.1088/0953-8984/21/39/395502>
47. Talirz L, Kumbhar S, Passaro E, Yakutovich AV, Granata V, Gargiulo F, Borelli M, Uhrin M, Huber SP, Zoupanos S, Adorf CS, Andersen CW, Schütt O, Pignedoli CA, Passerone D, Van de Vondele J, Schulthess TC, Smit B, Pizzi G, Marzari N (2020) Materials cloud, a platform for open computational science. *Sci Data* 7:299. <https://doi.org/10.1038/s41597-020-00637-5>
48. Choi C (2021) A web platform for scientific collaborations [Video]. YouTube. <https://youtu.be/2OnZ0DUigVA>. Accessed 10 Mar 2021
49. AltspaceVR <https://altvr.com/> Accessed 15 Aug 2021
50. NEOS <https://neos.com/> Accessed 15 Aug 2021
51. Glue <https://glue.work/> Accessed 15 Aug 2021
52. Rees GA, Kobayashi R and Moss VA (2021) The future of meetings: working in XR? [Conference session]. SIGGRAPH 2021, United States. <https://siggraph2021.hubb.me/fe/schedule-builder/schedule/sessions/868127>. Accessed 12 Aug 2021
53. SIGGRAPH (2021) Birds of a Feather. <https://thefutureofmeetings.wordpress.com/siggraph-2021-birds-of-a-feather/>. Accessed 15 August 2021
54. Viglione G (2020) A year without conferences? How the coronavirus pandemic could change research. *Nature* 579:327–328. <https://doi.org/10.1038/d41586-020-00786-y>
55. Roos G, Oláh J, Ingle R, Kobayashi R, Feldt M (2020) Online conferences—towards a new (virtual) reality. *Comput Theor Chem* 1189:112975. <https://doi.org/10.1016/j.comptc.2020.112975>
56. The Future of Meetings. <https://thefutureofmeetings.wordpress.com/>. Accessed 22 September 2021
57. Remmel A (2021) Scientists want virtual meetings to stay after the COVID pandemic. *Nature* 591:185–186. <https://doi.org/10.1038/d41586-021-00513-1>

58. Stahl A (16 April 2021) The future of offices and workspaces, post-pandemic. Forbes. <https://www.forbes.com/sites/ashleystahl/2021/04/16/the-future-of-offices-and-workspaces-post-pandemic/>. Accessed 29 July 2021
59. Brownlee D (2020) Twitter, Square announce work from home forever option: what are the risks. Forbes. <https://www.forbes.com/sites/danabrownlee/2020/05/18/twitter-square-announce-work-from-home-forever-optionwhat-are-the-risks/>. Accessed 31 July 2021

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.