Chapter 5

Building a Successful Computational Chemistry Laboratory

Jeffrey B. Schriber,[1] Dominic A. Sirianni,[2] George Flanagin,[3] Carol A. Parish,[4] and Bill R. Miller III[*,5]

[1]Department of Chemistry and Biochemistry, Iona University, New Rochelle, New York 10801, United States

[2]Department of Natural Sciences, Daemen University, Amherst, New York 14226, United States

[3]High Performance Computing, University of Richmond, Richmond, Virginia 23173, United States

[4]Department of Chemistry, Gottwald Center for the Sciences, University of Richmond, Richmond, Virginia 23173, United States

[5]Chemistry Department, Truman State University, Kirksville, Missouri 63501, United States

[*]Email: brmiller@truman.edu

A tenure-track position at a primarily undergraduate university (PUI) is highly rewarding, but can also be stressful with many new obligations and responsibilities that faculty were not necessarily prepared for during their graduate studies or postdoctoral fellowships. In particular, PUI computational chemists are likely to be the only computational experts in their department, and thus may not have access to experienced mentors to offer advice on establishing and maintaining a successful research group specializing in computational chemistry. In this chapter, we offer faculty beginning a tenure-track position advice and guidance on how to setup a young research lab. We describe several important considerations that faculty must contemplate as they start their labs. Furthermore, we detail different likely scenarios that faculty might encounter based on varying levels of financial, technical, and administrative support at a PUI and how faculty should proceed with establishing a new lab based on their specific situation. Finally, we provide advice for new faculty on how to recruit and train new students, as well as other important decisions faculty must make when trying to build a productive research group. This chapter provides a clear list of the most important aspects of creating a new computational chemistry research group to help new tenure-track faculty get started and be successful in the short- and long-term.

Parish and Hopkins; Physical Chemistry Research at Undergraduate Institutions: Innovative and Impactful Approaches, Volume 1
ACS Symposium Series; American Chemical Society: Washington, DC, 2022.

Introduction

Faculty typically begin tenure-track academic positions after completing their graduate work or a postdoctoral fellowship. New computational faculty have deep domain experience and were typically members of a research group in which computational resources were provided. A new academic position requires that the faculty member configure a research laboratory from the ground-up, substantially modify an existing computational space, or build a computational laboratory in a former experimental laboratory space.

Faculty must make many decisions ranging from the types of projects that they will direct, the computational tools they will employ, the hardware they need and the number of students they can mentor. These decisions in turn guide other decisions regarding the configuration, selection and installation of hardware and software, the configuration of the laboratory space and the distribution of available start-up funds.

These decisions are especially important when the new faculty member starts their academic career at a predominantly undergraduate institution (PUI). For PUI academic positions, the teaching loads are high (typically 5 courses per year, or higher), and research students are able to focus full time on research for only ~8-10 weeks in the summer. Ideally, research takes place year-round, but during the academic semesters the coursework of an undergraduate student must come first and take priority over research. This is in contrast to the research programs at R1 universities where advanced graduate students and postdoctoral fellows work full time on research year-round.

In this chapter, we outline some of the decisions that must be made when building a computational research laboratory and provide descriptions of possible configurations.

Initial Considerations

Many personal, professional, and financial factors must be considered when a PUI faculty begins a new research lab. Herein we detail several of the important factors that must be initially considered that help decide the future direction of a new computational chemistry lab.

Research Domain

The research domain of a future computational chemistry lab is perhaps the most important consideration in guiding purchase and setup of hardware. Traditionally, research in computational chemistry can be categorized as being mostly quantum mechanics-based (QM) or molecular dynamics-based (MD). For QM research, computers ideally have processors with high core counts and fast clock speeds coupled with access to large amounts of RAM (64+ GB) and high-speed storage (ideally SSD/NVMe). These workstations or compute nodes will typically have a new iteration of Intel Xeon or Intel X series CPUs with anywhere from eight to twenty-four (or more) cores. These core counts are based on 2022 hardware availability, and future CPU development historically leads to even higher core counts for low- and high- tier CPUs alike. Alternatively, the most important hardware component for a workstation designed for MD research is the graphics processing unit (GPU), which can easily be half the cost of a computer. MD workstations also need very large (4+ TB) amounts of high-speed disc to enable storage and analysis of very large trajectories associated with MD-based research. While some QM software does leverage the high-parallelism enabled by GPUs, notably TurboMole, most QM packages will benefit most from machines with high-powered CPUs. Presently, implementation of MD software is essentially universally done on GPUs. Likewise, recent machine-learning applications in computational chemistry also necessitate workstations with powerful GPUs if users intend to train their own models.

Parish and Hopkins; Physical Chemistry Research at Undergraduate Institutions: Innovative and Impactful Approaches, Volume 1
ACS Symposium Series; American Chemical Society: Washington, DC, 2022.

The prioritization of processing power on either CPUs or GPUs can have implications to the general setup of a lab. For example, computational labs whose research uses primarily MD will typically be composed of a number of individual workstations with GPUs and expansive disc space. Incoming faculty members at PUIs are likely unable to build an entire compute cluster of these machines, so individual workstations likely lead to a better setup for starting these types of labs. For QM research, a centralized compute cluster is somewhat more feasible, though still a very expensive, potentially prohibitive, option for incoming faculty. For both QM and MD-based labs, the use of external computing resources is extremely valuable, particularly for labs with access to limited computational resources on-site.

Principal Investigator Expertise

Just like incoming faculty at research-intensive (R1, R2, etc.) institutions, new PIs entering a PUI are domain experts in their respective field of research. The new PUI faculty member should understand that the institutional infrastructure and dedicated research support staff common at an R1 institution, likely exists at a different level at a PUI, and even among PUIs this can vary widely. PIs at a PUI must therefore often become jacks-of-all-trades in the various non-research technical skills which keep their research programs afloat and productive. While these skills are numerous and can vary widely with the level of institutional support provided, a few in particular seem to be common among PUI faculty, including:

- Systems engineering, whereby the PI is responsible for planning the design, setup, and implementation of their computing resources and systems,
- Systems administration, whereby the PI is responsible for the maintenance of software and hardware components necessary for continued successful use of computing resources, and
- Software development, whereby the PI designs and implements custom software tools to automate common tasks and reduce the technical load on undergraduate researchers.

Depending on the expertise of the PI at each of these (and the various other) non-research skills, certain decisions must be made to both maximize the research output of the lab and to provide the best experience possible for undergraduate researchers. For example, if the PI is not familiar with systems engineering and/or administration, then the lab will likely be better served to purchase prefabricated compute resources with an included manufacturer's warranty, rather than obtaining individual hardware components, assembling their workstations or clusters themselves, and performing their own hardware/software maintenance. If applicable, PIs should consult the Computer Science (CS) faculty and majors at their institution and inquire as to hiring a student part-time to help with these duties. Some departments with moderate funds may be capable of hiring a student part-time at a wage cheaper than a certified expert, or potentially as a scholarship obligation to the university for a CS student. This helps reduce the burden of new faculty addressing all hardware/software issues, while also giving a CS student valuable hands-on experience. Additionally, if the PI is not comfortable developing new software tools, the lab will likely be more productive by leveraging commercial software packages with dedicated helplines and a wide range of features than attempting to use and/or customize open-source package(s) with fewer (or potentially no) dedicated support staff focused on more specialized use cases.

Laboratory Design Elements

At the most basic level, computational research can be accomplished using a laptop and access to a centralized compute resource owned by the PI, the PI's institution, or national foundations (more details below) such as the NSF or DOE. As described above, for quantum work, and assuming adequate compute cycles are available, this may suffice. However, for MD work, it is necessary to invest in GPU-based workstations for simulation and the resulting computationally intensive analysis. For either the MD or QM case, it is ideal for research group *esprit de corp*, peer-mentoring and career-enhancing professional networking for the undergraduates to be co-located in a computational lab rather than distributed at desks in a library, or working remotely. In a computational lab, students will typically spend many hours at their desk, especially during the full time period of research during the summer. Ideally each desk is outfitted with a mouse, keyboard and one or more monitors. If students can connect to a centralized compute resource for their QM research, it may be enough for them to use their personal laptops. For MD research, desks should be equipped with powerful workstations that can handle visualization of long simulations.

Operating Systems and Software Requirements

Computational PIs may need to consider the choice of operating system (OS) to use in the laboratory. Most commercial and open-source research software, as well as software development tools and environments, run on the three major operating systems: Windows, OSX (run by Apple products) and LINUX. Even among scientists, many people are most comfortable performing their word-processing, spreadsheet and presentation tasks on software that runs on OSX and Windows (though, we recognize this may be changing with the increased adoption of next generation tools such as GitHub, LaTex, Jupyter, etc.) PIs typically begin their independent careers with an OS preference, although computational scientists may have experience with one or more OSs, or be OS agnostic. Typically, undergraduate students will own a laptop and have familiarity with either Windows or OSX. The PI's institution may provide support for some OSs but not others. It is also important to know how your software scales on the various OSs. These are all important variables that must be considered when building a computational lab. For instance, one of us began an independent career as a Mac-devotee for word processing and a LINUX user for research because our software ran significantly faster (on the same hardware) under the LINUX OS. Unfortunately, the institution supported only the Windows OS. Not wanting to be responsible for system administrative maintenance of both Mac and LINUX computers, the decision was made to switch the lab to the institutionally supported Windows platform for word processing/presentation.

Software Capabilities and Application-Dependent Hardware Requirements

Aside from the obvious considerations based on PI domain knowledge and desired software features and methodologies, several considerations exist for choosing appropriate computational chemistry software that affects user experience and overall research productivity, as well as the hardware necessary to use it successfully. First and foremost, among these considerations is the cost associated with licensing the software. Regardless of research domain, there exist both commercial and free software which may be either open- or closed-source. For example, software packages corresponding to the four combinations of these categories (from within the QM domain) are (i) Q-Chem, which is commercial and closed-source, (ii) Gaussian, which is commercial but open-source, (iii) ORCA, which is closed-source but free for academic research groups, and (iv) Psi4, which is free and open-source. Commercial packages in categories (i) and (ii) can cost up to $5,000

to obtain academic licensing for even a single research group (although discounts typically exist for purchasing licensing to multiple groups at the same university), however the initial expense is typically returned by an intuitive user interface and the offer of software maintenance and assistance after purchase. Free software in categories (iii) and (iv), on the other hand, are often written and/or maintained by academics (many of whom are graduate students or postdoctoral fellows), whose limited free time (among other considerations) prevents their investing significant time into either constructing an intuitive user interface or providing detailed software support for users. This may not be the case for all software in category (iii), as many such packages offer free licenses to academics but charge companies in the industrial sector, e.g., pharmaceutical companies or government agencies. In these cases, however, the "free" version of the software licensed to academics may be feature-redacted, as is the case for the PyMOL software from Schrodinger Inc., whose free academic version does not allow the user to ray-trace images for making publication-quality figures.

Differences in the initial cost and levels of software support available post-installation are not the only distinguishing features of these four categories of computational chemistry software, however, as the breadth of features included also varies widely between them. Commercial packages in categories (i) and (ii) tend to be "fully featured," with implementations of both widely used and more niche approaches included. Free packages in categories (iii) and (iv), on the other hand, typically specialize in one particular type of computation or target application, and include only the absolutely most common approaches (e.g., ground state Kohn-Sham density functional theory; KS-DFT in the QM domain) outside this chosen area. To illustrate this paradigm, let us consider an application in which QM-domain research labs may be interested: modeling electronically excited states of molecules. For simulating UV/Vis spectra in "medium-sized" molecules of up to a few hundred non-hydrogen atoms, the most widely used approach is time-dependent density functional theory (TDDFT). While TDDFT is available in *every* commercial QM code (*1*), several of the free codes either do not offer this functionality or have only recently added it. For a more detailed (and theoretically rigorous) examination of the energetics and properties of the excited states themselves, however, multireference (MR) formulations of closed-shell electronic structure methodologies such as second-order Møller–Plesset perturbation theory (MR-PT2), configuration interaction (MR-CI) or coupled-cluster (MR-CC) approaches are typically employed. A few of the most common packages used to perform these specialized computations are COLUMBUS, MRCC, Forte, PySCF, and CFOUR, each of which are free either for academic use (category iii) or free and open-source (category iv). Alternatively, among the most widely used commercial QM packages (Gaussian, Q-Chem, and Molpro), only Q-Chem and Molpro offer multireference CI functionality, and Molpro only offers multireference CC functionality by virtue of *including* the MRCC program within its distribution.

Even within a single research domain (QM or MD), the types of computations required by a particular application may require drastically different computational hardware to be utilized. For example, while nearly all QM-domain applications require a hefty CPU and a moderate amount of RAM, some may additionally require significant amounts of disk-based "scratch" space (upwards of 10 TB) for storing large intermediate arrays necessary for specific algorithm(s)/methods. Additionally, these types of computations may require billions of read-write operations in the construction and use of these intermediates, commonly referred to as "I/O" (for input/output); the primary examples of I/O-intensive computations are correlated wavefunction-based electronic structure approaches, such as CI, CC, and Moller-Plesset perturbation theory (MPPT). These computations require not only that large amounts of scratch space be available, but also that the disk drives be *local*, i.e., physically residing within the workstation/compute node itself, as network-mounted drives will drastically reduce computational efficiency due to network speed limitations

and/or traffic. Not only is this specific hardware setup not common practice among high-performance computing centers, (thereby limiting the usefulness of, e.g., XSEDE time; see below for further discussion), but even with access to a supercomputer with these nodes, performing these computations on an HPC resource typically requires that an entire node be requested for a significant amount of walltime, resulting in a potentially indefinite wait in the queue until the computation even begins. Therefore, I/O-intensive computations are much more amenable to being performed on dedicated, specialized workstation computers which the lab has acquired specifically for that purpose. Alternatively, density functional theory (DFT) computations are rarely I/O intensive, instead relying much more on available RAM and cross-core parallelization. Therefore, applications leveraging DFT computations are prime candidates for the exact brand of high-throughput quantum chemistry for which high-performance computing resources like XSEDE are designed.

Data Storage

Data backup and storage (short-term and long-term) is an essential function supporting a healthy environment in which all researchers will have confidence. It is a necessity, even if it need not be done in a specific way. Some characteristics of a successful backup strategy include:

1. **It must be capable of automatic operation.** Most processes can be run under the Linux cron or other scheduling software, but automatic operation is usually unaware of the login environment. This means the researcher must be able to supply the backup process with all parameters.
2. **It must also be capable of simple, *ad hoc* operation.** There are occasions when a researcher will want to backup an important result *now*.
3. **If the entire computer is being backed up, the process must run as root.** If only the "home" file system is backed up, the process still must have the ability to read all users' files.
4. **It must be runnable while the computer is in use.** While you may schedule it for midnight, much research goes on around the clock. This is called a hot backup as opposed to a cold backup that takes place when no other processes are running -- for example, before a major upgrade.
5. **The network connection to remote storage should not interfere with other network I/O.** Ideally, add a second network card to the computer so that bottlenecks are avoided by design.

Because backup can serve several different purposes, the purpose of the backup should be clear before a purchase is made. No single backup system can do it all. For example:

1. **If there is a need to recover *multiple* previous versions of a file,** then revision management software such as git can be used, and the git repos must be a part of the backup for this to be effective.
2. **If one is guarding against hardware (disc) failure,** then duplication can be pushed down into the hardware level with RAID 1, 5, 6, 10, or 60 configurations involving several physical discs.
3. **If one is concerned that the lab itself might be destroyed** in an earthquake or fire, then offsite backup is clearly required.
4. **If the data are sensitive,** and there is fear of misuse or theft, then encryption should be employed.

Many of the preceding properties interact counterintuitively. As an example, a large RAID array *does* protect the data from the failure of any one (or two) discs, but merely by having many discs the probability that one of them will fail increases no matter how reliable they are. Additionally, there is the risk that if a disc is failing due to a cause that is exogenous, the risk of additional failures in the same array increases. And finally, hot-swap discs (replacement of the hardware while the computer continues to run) can cause long delays in return to normal operation, and the delay is made longer when striping and parity are involved, which it is in all cases except RAID 1 and RAID 10.

Scenarios for Incoming PIs Starting at a PUI

New faculty at PUIs will find themselves in a unique situation based on the resources (financial, lab space, support staff, etc.) available at their institution. These resources will likely dictate the possible directions their lab will undertake, particularly in the early years of their research group. Herein, we discuss the most common scenarios faculty will discover upon arriving at their new institutions, and how the available resources can be best utilized to help set up a productive and successful research group.

Significant Financial Investment with No Major Lab Space Limitations

If a new computational faculty member is lucky enough to have sizable institutional start-up funding, access to a local high performance cluster, system administrative support and a generously sized laboratory space, they can focus their resources on research personnel (student support in the summer or academic year, post-baccalaureate or post-doctoral fellow, research technician) and equipping the lab space with turn-key workstations. This scenario is more common at research-one institutions, however funding at some research intensive PUIs can approach this level of support.

It is ideal to train fledgling undergraduate computational scientists in the use and mastery of the LINUX operating system, command line control of computational applications, scripting and the use of text editors such as Vim. Undergraduates also need to learn how to analyze data using LINUX-based research software, and to communicate their results with both LINUX and non-LINUX software tools. For this reason, it is ideal for each student desk in a computational chemistry laboratory to have a word-processing laptop or desktop (PC, Mac or LINUX) as well as a powerful LINUX workstation. For quantum projects, the LINUX workstation can be considered optional - while it is helpful to be able to quickly run test calculations on a local resource - students can run their calculations on a cluster and use the cluster to master LINUX; assuming there is a cluster available with adequate compute cycles. However, for molecular dynamics projects a powerful LINUX workstation with onboard GPUs and ample memory is critically important for the computationally intensive task of analyzing molecular trajectories. For any type of computational project, it is ideal to have at each desk a large screen monitor or a workstation capable of outputting a display to multiple monitors. This allows reading multiple files simultaneously and minimizing scrolling, window placement and resizing, and enhances research analysis and productivity.

If resources are available for research personnel, this type of expenditure can be very beneficial for a research program as well as the students supported. At many undergraduate institutions, research takes place all year, but the summer is a time of especially focused effort. During the summer, undergraduate students can focus their full-time effort on research, and if the PI can develop a lab culture that encourages students to participate for multiple summers and the intervening academic semesters, such efforts can lead to publishable results. Having a peer-reviewed publication as an undergraduate will help students in their post-baccalaureate goals; whether those

goals include graduate or professional school placement, employment or high school teaching. Regardless of eventual career, participation in outcome-oriented undergraduate research demonstrates their ability to see a project from inception through to completion. At many institutions, students can sign up to do research during the semester for graded or pass/fail course credit. This is an excellent way for students to carve out time to keep their research progressing during the academic semester, and to show (via their course transcript) their continued involvement. However, some students must have a paid job during the academic semesters in order to support their education. In these cases, it is ideal to pay these students as hourly researchers, to avoid them having to juggle classes, a paid job AND research. Having the option of performing research for course credit or pay allows all students the option of academic year research and supports institutional goals for equity and inclusion.

Using funds to hire recently graduated research students as post-baccalaureate fellows supports students and the research enterprise. Increasingly, students will work in the same laboratory for multiple years and become highly trained collaborators. Often these students are interested in continuing their work in the lab as a post-baccalaureate fellow either for the summer following graduation, or (less frequently) for a gap-year experience or employment. Spending time as a post-baccalaureate fellow allows students time to complete publications and gain experience mentoring less experienced students which is good for their professional development. Finally, a good use of personnel funds is to hire a postdoctoral fellow (PDF). This is an ideal position for someone who is planning to pursue a tenure-track position at a predominantly undergraduate institution as it gives them experience mentoring students in outcome-oriented research, designing projects that are doable by undergraduates, and juggling the often competing demands associated with such a position. It allows them to have direct experience with the position that they seek, and to develop the time management skills necessary to be successful in their independent career. It is important to mentor the PDF in their expectations of undergraduates as the PDF will be focusing on research full-time, year-round whereas undergraduates can only focus their full attention during the summer. While there is no one-size-fits-all approach to designing a research project for a postdoctoral fellow (PDF) that will involve undergraduate students, it is important to discuss with the PDF various design considerations including the difficulty or complexity of the project, the estimated timeline to completion, the number of students on a project and strategies for student training, record keeping and communication. Regular check-ins and goal setting will allow the PI to mentor the PDF, and allow the entire team to keep the project on track. Another useful collaboration between the PI and PDF is to participate in shared, team teaching, but this must be approached with caution as the most important aspect of the experience for the PDF is to generate publications demonstrating that they can manage time, design new projects, mentor students, and produce independent scholarship. Having a postdoctoral fellow in the lab is also good for undergraduates as this situation increases mentor opportunities and gives students access to someone with different experiences and expertise, including the most recent computational technology.

Significant Financial Investment with Lab Space Limitations

New faculty that find themselves with a small dedicated research lab at a PUI, but with plenty of startup funds have several options at their disposal despite the limited physical space. The lack of available space precludes the faculty member from maintaining many individual workstations in the lab. Thus, students will likely have to share workstations (see below) within the lab space itself, which means access to alternative compute cycles would become necessary. If enough funds are available (likely in excess of $100-200K), the natural choice would be to purchase a compute

cluster that is ideally stored and maintained by the institutional Information Technology (IT) staff. Additionally, funds could be used to hire a [part-time] technical support person with dedicated duties that include installing new hardware/software and maintaining the research computer cluster. If funds are not sufficient for purchasing a cluster, then requesting and gaining access to an external high-performance compute cluster would be critical for establishing a productive research group at a PUI with limited lab space available. Such external computing resources include those available either directly through, e.g., the National Science Foundation's Extreme Science and Engineering Discovery Environment (XSEDE) or the Department of Energy's Advanced Scientific Computing Research (ASCR) programs, or via membership of one or more of several computational chemistry research consortia, including the Molecular Education and Research Consortium in Undergraduate Computational Chemistry (MERCURY) (*2*), the Midwest Undergraduate Computational Chemistry Consortium (MU3C) (*3*), or Molecular Computation and Visualization in Undergraduate Education (MoleCVUE) (*4*).

The limited lab space prevents the use of many individual student workstations, and thus the workstations that are obtained for the lab are 1) intended to be shared by multiple research students, and 2) should be powerful enough to perform all calculations and analyses that cannot be carried out by the remote compute cluster. The number of and specifications for the workstations will depend on the nature of the research being performed, the number of students expected, and the number of workstations that can functionally fit in the lab. Even with limited space, faculty should look to have a minimum of 2-3 workstations for students on-site. Having even just a few workstations allows multiple students to work simultaneously, and it also enables more experienced students to train new lab members. In this situation, PIs may want to encourage students to use their personal computers/laptops, when possible, especially when accessing remote clusters to perform calculations and analysis. This would give the lab workstations priority for students who may not have reliable personal computers, and students that need specialized software or hardware for calculations/analyses that may only be available on the lab workstations. Faculty may also set up a schedule (or sign-up sheet) for students to select available times to use the research workstations located in the lab.

With sufficient funds, faculty with limited space have the option to purchase high-end, pre-built workstations from respected vendors for their research needs. For groups that perform MD simulations regularly, faculty should typically look to purchase workstations with 2-4 GPUs each, ideally maximizing the number of GPUs per computer that is affordable. These pre-built, optimized GPU workstations, depending on the specifications of the hardware (e.g. CPUs, GPUs, etc), could cost $5,000-$15,000 each. Multi-GPU workstations running MD often generate substantial heat, so faculty should be mindful to get a larger computer case with significant cooling ability (e.g. liquid cooled CPUs and/or GPUs). This is especially important for faculty with limited research space because small rooms with GPU workstations tend to be very warm, and could become uncomfortable for students and unsafe for the hardware itself with prolonged exposure to high temperatures. Additionally, any faculty with a computational lab in a relatively small room, should consider their available options for keeping the room cool and well-ventilated. Additionally, depending on the storage needs for the group, faculty could consider setting up their workstations with a RAID array or similar configuration (see backup considerations above for more information). For groups that utilize QM calculations, computers generally cost less and faculty will want to spend more resources including high-performance CPUs, more RAM, and plenty of available storage space into each workstation. Cooling considerations are more easily accommodated because CPUs do not produce as much heat as GPUs, and water-cooled CPUs are more commonplace and affordable, (compared to GPUs). Overall, faculty with relatively small research spaces and moderate financial

support have options available to them, but most options rely heavily on balancing a small number of physical workstations in the lab space with students regularly making use of remote resources.

Reduced Financial Investment with No Major Lab Space Limitations

Another scenario for incoming faculty is to have a dedicated research lab space, and a small startup package. While construction of a high-powered compute cluster from scratch is likely unfeasible (i.e. financially prohibitive), the dedicated space allows for a number of viable lab configurations depending on research domain, institutional support/existing hardware, PI expertise, and operating system/software requirements. In all cases, labs starting with limited funds and a reasonable amount of space will typically be composed of a set of high-powered workstations rather than a centralized cluster. Additionally, use of external compute resources mentioned above (e.g., XSEDE, ASCR, MERCURY, MU3C, MoleCVUE, etc.) is also important particularly for cases where many high-throughput computations are necessary. Ideally, due in part to available space, labs can offer at least one workstation per student, supplemented with compute resources either a) from existing compute nodes, b) from compute nodes shared within the campus, and c) compute nodes from external organizations.

This multi-level approach to computation allows for a variety of setups. For example, startup money can be allocated so that each student gets an inexpensive (potentially free), low-performance desktop from which to access a number of different resources, including lab-purchased workstations and internal and external compute clusters. Remaining funds can be spent on these high-performance workstations, or potentially compute nodes to be added to an existing cluster, all of which is shared among students. For larger startup packages (and/or smaller labs), all students could be provided their own high-powered workstation, from which they could also access any other compute resources available.

Certainly, the specifications and cost of the in-house workstations depend heavily on the nature of research being performed. For computational labs that mostly (or even partly) use MD as a main research tool, startup money is best spent on as many workstations as can be afforded. These workstations can be anywhere from around $5,000 up to $15,000 each, where roughly half of the cost of the machine can be due to a necessary high-capacity GPU. Depending on the budget, it is possible that only a few machines can be bought. In addition to the GPU, a lot of storage is needed, ideally fast storage in the form of high-capacity SSDs potentially with NVMe interfaces. The most economical way to get these workstations is to build them from scratch, but some degree of PI expertise is required. QM workstations tend to be cheaper than MD workstations since high-performance CPUs tend to cost significantly less than GPUs. Furthermore, even mid-tier Intel chips (for example, core i7s) can be serviceable due to their fast clock speeds and moderate numbers of cores (6-12 cores, typically), all being currently available for anywhere from $200 to $600 depending on core counts. The other important resource for a QM workstation, RAM, can be bought for roughly $250 per 64 GB, though fluctuations in price can occur. In sum, a 10-core QM workstation with 64 GB RAM, 2 TB of NVME SSDs, and 8 TB of storage on hard drives can be built for roughly $1,800 (in 2022). Importantly, prices and available hardware specifications are intimately tied to when they are purchased. The example workstation purchase we provide here would certainly look different in ten or even five years, with the CPU, memory, and storage likely being both faster and higher-capacity. What remains the same is that the quality and capacity of CPU and RAM should be prioritized in a QM workstation, and thus they will be the most expensive components in typical builds. With ample space, numerous workstations similar to the one mentioned can be used to get students starting with running jobs and obtaining publishable results.

For both QM and MD projects, a lab can be started by students sharing however many workstations are affordable, whether by physically sharing the time, or by accessing them remotely on inexpensive desktops. A benefit to this model is in its extendibility; workstations are easily upgradable, and certainly more can be bought with additional funding. A room full of desktop computers, with as many configured as powerful workstations as possible, coupled with access to additional campus-wide or external compute resources, allows a new research lab to get publishable results on many projects. Ideally, this productivity can lead to additional grant funding for either more MD/QM workstations and/or a local, lab-maintained compute cluster.

Reduced Financial Investment with Lab Space Limitations

For new faculty entering a small university, or one which places less emphasis on research excellence relative to teaching and service, the most likely scenario is one in which neither a significant startup package is offered nor a dedicated research laboratory space (or, at least, a limited one) provided. In this scenario in particular, these institutional factors play a significant role in determining both the laboratory setup and overall research environment, regardless of domain (i.e., MD vs. QM). The most space- and cost-efficient solution is likely for undergraduate researchers to use their personal laptop computers to access the remote computational resources mentioned above, this setup erects an unnecessary barrier excluding undergraduate researchers who do not have the financial means to own a powerful laptop. A more equitable and inclusive solution would be to obtain a set of reasonably powerful and stable laptop computers for the lab which students may "check out" for the semester or summer research period(s), from which students may access remote computational resources and perform data analysis/visualization. One possible source for these laptops would be from the institution's own inventory surplus or refurbished supply, if available, or alternatively the PI could purchase them using their (albeit limited) startup funds. Depending on the research domain, a reasonable laptop might be equipped with a basic Intel i5 or i7 CPU, up to 8 GB of RAM, 500GB of disk space (likely HDD and not SSD), and a basic graphics card rather than an on-board GPU. Such a laptop could be purchased from one of several vendors for $400-$600 each, allowing for even large labs (>=10 students) to furnish a laptop for each undergraduate researcher while leaving some startup funds left over. Fortuitously, maintaining a set of "labtops" (lab laptops) also offers the additional benefit of being able to specifically curate the software available on the labtops, ensuring a uniform experience for all undergraduate researchers and limiting issues related to incompatible workflows and/or software versions.

Even if the problem of space is solved by using external compute resources and providing laptop computers for use by the undergraduate researchers, it will likely still be necessary for the lab to maintain at least one local workstation computer in order to perform tasks which are not efficient uses of compute cycles on remote HPC resources, e.g., analyzing MD trajectories or performing I/O-intensive QM computations. As discussed above in Scenario 3, the system, hardware, and software requirements for in-house workstations — and therefore their price — can vary widely based on research domain. If the PI has sufficient systems engineering/administration experience, they may choose to build their own workstation(s) from individual hardware components to minimize their costs, however they must balance this up-front savings against the "sweat equity" required to maintain their hardware and software components themselves. Similarly, if the PI has experience developing custom software, they may forgo the purchase of commercial packages for data generation and/or analysis, with the understanding that they are then responsible for their own troubleshooting.

Long-Term Considerations

Faculty at PUIs that specialize in computational chemistry research will undoubtedly have many issues to consider to maintain sustainable and productive research groups for their long-term success in the field and to provide enriching experiences for undergraduates in their lab. Herein we detail several topics that many faculty encounter with their research groups.

Time Management and Saying "No"

As this chapter outlines, new faculty must juggle many demands on their time. There is an exhilaration that comes from landing a tenure-track position, and being welcomed into the academic world. There will be a temptation to attend every reception to meet your colleagues and hob-nob with the Dean or University President. You might be inclined to volunteer for any and all important committees. Keep in mind that to achieve tenure, you must meet or ideally exceed the institutional expectations which often heavily emphasize excellent teaching and student mentoring. And while it is important to attend social functions to build an on-campus network, and to serve on some committees (ideally ones that you are passionate about and/or student-related), no institution that we know of weights these activities more than, or in lieu of, excellence in teaching and scholarship. It is important to learn the art of a respectful "no" when asked to participate in tasks that detract from the main components of your work. It is also important to manage your time well, and to focus your efforts on those tasks that will be emphasized at the time of tenure and promotion.

Recruiting Students

Recruiting students to work on a research project is not an activity that many faculty are formally prepared for during graduate school or a postdoctoral experience, however it is a necessary part of starting a new research group. Many faculty naturally recruit students from classes they teach by casually mentioning their research and specialties as it relates to certain topics during lectures. Additionally, new faculty should advocate for presenting at a departmental seminar that is advertised to students interested in the chemical and computational sciences. New faculty can also ask to briefly visit with certain classes (such as seminar classes for majors and introductory level chemistry courses) targeting the intended student demographics. Student organizations (e.g. ACS, Alpha Chi Sigma, ASBMB, etc) are also a good resource from which faculty can recruit new students, including presenting a research talk at a student organization's general meeting (if allowed).

New faculty are encouraged to recruit students from early and later in their undergraduate studies; students further along in their major will usually pick up the skills and material quicker since they generally have a more solidified understanding of the background material, but students earlier in their studies will have longer tenure in the lab. Students who start early and persist become highly experienced, and can help with training newer students (see more below about training students). Computational chemistry research is more amenable to recruiting less experienced students since there is essentially no safety risk to students involved in research as a first-year or sophomore, even if they have not been exposed to the proper techniques or background information for the lab (unlike many wet lab settings). Unlike graduate students, undergraduate students may not be motivated entirely by a desire to learn about the world around them, and thus may desire to see a clear real-world purpose for a project. This purpose can aid in understanding a project, and build passion and motivation to complete the project. Thus, new faculty should be prepared to be flexible and willing to change, modify, or add new research projects that might be more clearly relevant to undergraduates to increase recruitment.

Parish and Hopkins; Physical Chemistry Research at Undergraduate Institutions: Innovative and Impactful Approaches, Volume 1
ACS Symposium Series; American Chemical Society: Washington, DC, 2022.

In some cases, newer faculty might find themselves overwhelmed by the response from students interested in research (especially if they have recruited well) and be tempted to allow all students into their new research group. One way to address lab over-subscription would be to establish requirements for entering the lab, such as only allowing juniors or seniors to join. As an alternative approach, PIs can devise an entrance quiz for students to join their research group based on the basic background or skills needed for the lab. The quiz is not intended to be particularly difficult or challenging (in fact, it could be repeated if a student does not perform well). It is also not intended to prevent underprepared students access to research, but instead could be used to slow the recruitment of students into a lab (if necessary). In practice, the role of the quiz should be to expose students to the knowledge they will need to be successful in the research group, and ensure that students understand the type of research they are signing up for before committing themselves to a research lab. As an example for this approach, a computational biochemist whose group runs MD on proteins could give students an entrance quiz where students must know the names, letter codes, and structures for all twenty amino acids, in addition to demonstrating competency with a dozen (or so) of the most common Unix commands (e.g. ls, pwd, ssh, etc) using a command line interface. An entrance quiz allows a PI to more easily control the size of their group, while also helping ensure that students joining their group are willing to put in the effort necessary to excel. Additionally, students will have already learned some of the basics of research before they begin their work, which makes training them more efficient.

Training Students

Once a new PUI faculty has recruited students, the process of teaching them how to be productive research students begins. Even with a small number of students, new groups should establish a regular (weekly or biweekly) group meeting time that can be used for the PI to present background information on specific projects or background information (e.g. the basics behind the math of quantum mechanics). As students begin to perform calculations and generate results, they should also get into the rotation of presenting their work to the group. Routine meeting times set a precedent for regular communication between faculty and students where each are able to ask questions in an informal environment. As a research group grows larger, PIs should also have [less often] regular meetings with the members working on each project to get specific updates from students, and set goals (both short-term and long-term) for each student on what they plan/hope to accomplish before the next meeting.

Initially (when there are only 1-4 students in the group), faculty should spend significant one-on-one time with each student teaching them how to perform basic research tasks (including extensive LINUX commands and standard QM/MD calculations), while also providing clearly written protocols the students can use for reference when working individually. Faculty should also provide literature for students to read, though preferably start with access to review papers that may be easier to understand for new students entering the field (especially important for less experienced students). This method of personal training should continue until these students have grown confident in their abilities, while faculty should encourage students to utilize other resources (such as software manuals and internal protocols, and more experienced students in the lab) prior to asking the faculty for help. The PI should especially encourage students to consult one another, because while one student is learning the other is becoming more comfortable with their knowledge and growing as a potential mentor for future students.

This style of one-on-one, hands-on mentoring is not sustainable long-term for [new] faculty given the high teaching loads at PUIs. Thus, once 2-4 students are fully trained (this goal is different

for each faculty and their own individual research group), each new student that joins the group is assigned a mentor (initially one of the original 2-4 fully trained students) that will teach the new student(s) the same methods, techniques, and background information that were taught to them by the PI. This "scaffolding" approach to training should make an undergraduate research lab sustainable long-term so long as the students buy-in to the process and embrace taking on the role of a responsible mentor for the new students. PIs should also make it clear to new students that they are always welcome to contact the faculty with questions, concerns, and issues, but that they are expected to at least inquire with another student (preferably their mentor) beforehand. PIs are also encouraged to set up a group communication forum (e.g. GroupMe or Slack) so that students can ask questions and make comments to everyone in the group, thus allowing students to support one another when they are not meeting in-person. Faculty should make themselves available to students in a variety of communication methods (e-mail, text, GroupMe, etc), though clear boundaries (i.e. times of availability, modes of communication, etc.) should be established based on personal preferences of the faculty so as not to allow their time and life to be consumed by questions and issues from students.

The Importance of Seeking External Funding (Including Compute Cycles)

As illustrated above, there are many tasks that a new faculty member must devote their time and attention to, including fundraising to support their research ideas and their laboratory. The timing of this effort is dependent on the start-up resources provided and the institutional expectations. There are a handful of programs that support early career scientists with well-defined application windows, i.e. first 3-5 years of a tenure-track position. These programs include the Research Corporation Cottrell Scholars award (*5*), the American Chemical Society Petroleum Research Fund Undergraduate New Investigator grants (*6*) and the National Science Foundation CAREER awards (*7*). In those cases, it is ideal to have a plan to submit at least twice during the application window so that the application can benefit from reviewer feedback. One strategy for successful external grant-seeking is to begin by applying for smaller grants made available from private or regional foundations. Lessons learned and preliminary results gleaned from these efforts can be used to seek larger grants from federal agencies such as the National Science Foundation, National Institutes of Health, Department of Energy, etc. Grant-seeking can be a time-intensive and sometimes frustrating task, but the intellectual freedom that comes with external funding allows significant advancement of the individual PI's research program. Another source of research support for computational investigators can be found in shared national resources such as the Extreme Science and Engineering Discovery Environment (*8*) or the Department of Energy's Innovative and Novel Computational Impact on Theory and Experiment (INCITE) program (*9*). Both resources have extensive online documentation and training materials. For the XSEDE program, start-up allocations for new investigators are relatively easy to obtain, requiring simply an XSEDE account, abstract and CV. For both XSEDE and INCITE, larger allocations are awarded on a competitive basis. It should also be noted that XSEDE ends on August 31, 2022 and will be replaced by a new program called ACCESS (Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support) starting on September 1, 2022, so PIs will be requesting compute cycles from ACCESS starting in fall 2022.

Conclusion

Securing and starting a tenure-track position at a primarily undergraduate institution is an exciting and overwhelming experience that is highly rewarding, especially when working with students in a computational chemistry research setting. New faculty have many items to consider,

such as their own expertise, experience, and ability that will drive the direction of their nascent research program. Any available physical, financial, and technical support resources will have a major impact on the types of research that can be pursued and the style of lab that can be designed and implemented. Herein we have described the most likely scenarios that new faculty will encounter, including suggestions on how to proceed depending on the research domain (QM vs MD), PI expertise, and available assets. Finally, we offer general advice for new faculty on how to maintain a long-term computational chemistry research group at a PUI, including guidance on recruiting and training students, as well as how to choose and handle software and hardware choices. It is our genuine hope that this chapter will help provide suggestions and better prepare new tenure-track faculty aspiring to build a successful computational chemistry research group with undergraduates.

References

1. Category:Computational Chemistry Software. *Wikipedia*; 2020.
2. Shields, G. C. The Molecular Education and Research Consortium in Undergraduate Computational Chemistry (MERCURY): Twenty Years of Exceptional Success Supporting Undergraduate Research and Inclusive Excellence. *Scholarsh. Pract. Undergrad. Res.* **2019**, *3* (2), 5–15.
3. Kuwata, K. T.; Kohen, D.; Krueger, B. P.; Polik, W. F. Developing a Regional Computational Chemistry Consortium through Undergraduate Research Conferences. *Counc. Undergrad. Res. Q.* **2012**, *32* (4), 9–15.
4. *MoleCVUE*. https://sites.google.com/view/molecvue/events (accessed 2022-08-16).
5. *Cottrell Scholar Award*. https://rescorp.org/cottrell-scholars/cottrell-scholar-award (accessed 2022-08-16).
6. *Undergraduate New Investigator (UNI) Grants - American Chemical Society*. https://www.acs.org/content/acs/en/funding/grants/petroleum-research-fund/programs/undergraduate-new-investigator-grants.html (accessed 2022-08-16).
7. *Faculty Early Career Development Program (CAREER) | Beta site for NSF - National Science Foundation*. https://beta.nsf.gov/funding/opportunities/faculty-early-career-development-program-career (accessed 2022-08-16).
8. *Home - XSEDE*. https://www.xsede.org/ (accessed 2022-08-16).
9. *INCITE – Leading the way for Scientific Computing*. https://www.doeleadershipcomputing.org/ (accessed 2022-08-16).