Atomic and Molecular Scattering Applications in an Apache Airavata Science Gateway

Barry I. Schneider
barry.schneider@nist.gov
National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

Klaus Bartschat
Oleg Zatsarinny
Kathryn R. Hamilton
klaus.bartschat@drake.edu
oleg_zoi@yahoo.com
kathryn.hamilton@drake.edu
Drake University, Des Moines, IA 50311, USA

Igor Bray
igor.bray@curtin.edu.au
Curtin University, GPO Box U1987, Perth, WA, Australia

Armin Scrinzi
armin.scrinzi@lmu.de
Ludwig-Maximilians Universität, D-80333, München, Germany

Fernando Martín
Jesús González Vázquez
fernando.martin@uam.es
jesus.gonzalezv@uam.es
Universidad Autónoma de Madrid, Catoblanco, Madrid 28049, Spain

Jonathan Tennyson
j.tennyson@ucl.ac.uk
Department of Physics and Astronomy, University College London, Gower Street, London, WC1E 6BT, United Kingdom

Jimena D. Gorfinkiel
jimena.gorfinkiel@open.ac.uk
School of Physical Sciences, The Open University, Milton Keynes MK7 6AA, United Kingdom

Robert Lucchese
rlucchese@lbl.gov
Chemical Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

Sudhakar Pamidighantam
pamidigs@iu.edu
Cyberinfrastructure Integration Research Center, Pervasive Technologies Institute, Indiana University, CIB 2709 E 10th Street, Bloomington, IN 47408, USA

ABSTRACT

We document recent progress made in the development and deployment of a science gateway for atomic and molecular physics (AMP) [10]. The molecular scattering applications supported in the gateway and the early phase of the project were described in a preliminary report [33]. Here, we present recent advances in both the platform's capabilities and in its adoption for additional software suites and new possibilities for further development. The past year has seen substantial progress, with the addition of two new software suites and additional authors. A very successful workshop, supported by the MOLSSI, NSF, and NIST, was held at NIST from Dec 11-13, 2019. The agenda contained discussions of the science as well as demonstrations of the codes both in production and learning modes. More than 30 scientists participated in the workshop. Over the past few months, the number of registered gateway users has grown to over 60.

The applications being deployed provide users with a number of state-of-the-art computational techniques to treat electron scattering from atomic and molecular targets, as well as the interaction of radiation with such systems. One may view all of these approaches as generalized close-coupling methods, where the inclusion of electron correlation is accomplished via the addition of generalized pseudostates. A number of the methods can also be employed to compute high-quality bound-state wave functions by closing the channels and imposing exponentially decaying boundary conditions. The application software suites are deployed on a number of NSF and DoE supercomputing systems. These deployments are brought to the user community through the science gateway with user interfaces, post-processing, and visualization tools. Below we outline our efforts in deploying the Django web framework for the AMPGateway using the Apache Airavata gateway middleware, discuss the new advanced capabilities available, and provide an outlook for future directions for the gateway and the AMP community.

PEARC '20, July 26–30, 2020, Portland, OR, USA

ACM ISBN 978-1-4503-6689-2/20/07…$15.00
https://doi.org/10.1145/3311790.3397342

CCS CONCEPTS

• **Applied computing** → **Physics**; **Chemistry**; • **Human-centered computing** → **Accessibility systems and tools**; • **Computing methodologies** → Massively parallel and high-performance simulations; Quantum-mechanical simulation; Molecular simulation; Scientific visualization.

KEYWORDS

Atomic and Molecular Physics, Science Gateway, Light-Matter Interaction.

ACM Reference Format:
Barry I. Schneider, Klaus Bartschat, Oleg Zatsarinny, Kathryn R. Hamilton, Igor Bray, Armin Scrinzi, Fernando Martín, Jesús González Vázquez, Jonathan Tennyson, Jimena D. Gorfinkiel, Robert Lucchese, and Sudhakar Pamidighantam. 2020. Atomic and Molecular Scattering Applications in an Apache Airavata Science Gateway. In *Practice and Experience in Advanced Research Computing (PEARC '20), July 26–30, 2020, Portland, OR, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3311790.3397342

1 INTRODUCTION

Based on enthusiastic participation of several groups after the 2018 ITAMP workshop on "Developing Flexible and Robust Software in Computational Atomic and Molecular (A&M) Physics" [34], a concerted effort began to organize the software from this group of internationally recognized researchers in computational atomic and molecular physics, to identify and prioritize outstanding problems in A&M science that would benefit from a wider deployment of these software tools and algorithms, and to disseminate these applications in a common framework that would lead to more rapid scientific progress for the entire AMP community.

This effort is being undertaken to ensure that these codes, some representing decades of effort, can continue to produce high-quality scientific results for a broader community and be usable by upcoming scientists. In many cases, these codes were developed by graduate students and postdocs within particular groups, and are not well documented. If they are to be useful in the long term, that must be remedied. In addition, we view these codes as dynamic entities that will change in time to incorporate new physics and new and better programming models and practices. The tools, however, have been demonstrated to be capable of treating scientific and technologically interesting problems, often using sophisticated physical models. This strongly motivates our efforts.

The gateway is currently deployed via the NSF XSEDE project [16]. The group benefitted from an XSEDE startup award, which included ECSS support, and from additional computational resources from Directors Discretionary allocations from the SDSC and TACC HPC centers. We are indebted to both centers for their support and broad scientific vision. In the past year, a number of the codes were ported and are running on various XSEDE platforms. In a recent workshop [11], the codes were described in detail to a larger AMP community and demonstrations were provided on how they are deployed within the AMPGateway environment. Some progress has been made in making them usable by others outside the local groups. We are now taking steps to do this on a much larger scale.

Technically, the AMPGateway is powered by the multi-tenanted Apache Airavata middleware framework [9, 30] and served by SciGaP hosting services for sustained operation. In the first stage of our efforts, the software suites were deployed as independent applications with specific input interfaces. Community building has already started, and a few additional software suites were identified for inclusion in a second phase. The interoperability of the software suites is very important and will be addressed as a follow-up.

As stated previously, the present manuscript is intended as an update to the earlier paper [33]. We encourage readers to refer to that article for details. In Section 2, we present brief descriptions of the seven AMP codes that are already available on the gateway. Section 3 is devoted to the details of the construction and deployment of the gateway. In Section 4 we discuss issues of broadening usage of the gateway and questions of community building.

2 APPLICATION SOFTWARE SUITES

We have concentrated our major effort to date on deploying five codes on the AMPGateway. Recently, two more codes were added to the mix. Very brief descriptions are given below and a summary appears in Table 1.

2.1 BSR

The B-spline R-matrix (BSR) method and the accompanying computer code [40] were developed by Zatsarinny in the Bartschat group at Drake University. The program computes transition-matrix elements for electron collisions with atoms and ions as well as photoionization processes. From these, cross sections and other experimentally observable parameters can be obtained. The code can also be run in a mode that provides atomic structure information through energy levels and oscillator strengths.

The BSR approach is a particular variant of the R-matrix method to solve the close-coupling equations in coordinate space. It is an alternative formulation of the well-known R-matrix code developed in Belfast under the long-term leadership of Philip Burke [8]. It is also complementary to the convergent close-coupling (CCC) approach described below. The published BSR code [40] is a serial version, which was written in the non-relativistic and semi-relativistic (Breit-Pauli) frameworks. A comprehensive overview of the BSR method and its applications may be found in [41].

2.2 CCC

The original implementation of the CCC method was designed to produce accurate cross sections for scattering of light projectiles from quasi one- and two-electron targets [7]. It began with electron-hydrogen scattering [6], and was further extended to quasi-one electron targets that could be modeled well by one valence electron above a frozen Hartree-Fock core [5]. It was then extended to the helium target [19], and quasi two-electron targets such as Be [20]. The main features are:

- Expansion of the target state in a sufficiently large $\mathscr{L}^2$ basis size N to treat cases where both excitation and ionization of the target are possible, with convergence tested by systematically increasing N.
- Expansion of the scattering wave function in the momentum based Lippmann-Schwinger equation.
- Introduction of numerical quadrature to reduce the problem to a very large set of linear algebraic equations.

The CCC codes have been parallelized to use OpenMP on a node and MPI between nodes. They have been deployed successfully on Comet and Stampede2. A GPU implementation, currently under development, has already shown great promise with a speed-up of up to two orders of magnitude.

2.3 UKRMol+

The UK Molecular R-matrix codes were designed to treat low-energy elastic and inelastic electron-molecule collisions using the

R-matrix method; they have evolved to study photoionization and positron-molecule collisions as well as to produce the input required for time-dependent molecular R-matrix with time (RMT) calculations [13]. Similar to BSR they are based on the R-matrix method. The general theory is described by Burke [8].

The now frozen release version of the (mainly serial) UKRMol suite uses Gaussian Type Orbitals (GTOs) to represent both the target and continuum orbitals [15]. A review article by Tennyson [39] gives a comprehensive overview of the theory used and of the code's functionality.

A newer version of the code [28] has been developed led by Mašín and Gorfinkiel. This work has improved the suite's overall efficiency and relaxes previous limitations on the size of the inner region and energy range by augmenting the continuum basis with a set of B-splines. The code uses the new GBTOlib integral library to determine all the one- and two-electron integrals needed in the mixed basis of GTOs and B-splines.

In addition, there is a rich array of outer-region functionality, including automated resonance detection and fitting, bound-state detection, computation of multichannel quantum defects, rotational excitation, and photoionization cross sections. A full release of the suite UKRMol+, regularly updated, is available on [42].

2.4 tRecX

The tRecX code package [14, 37] is a general framework for solving initial-value problems of the form

$$\frac{\partial}{\partial t}\Psi = \mathcal{D}[\Psi, t] + \Phi(t) \tag{1}$$

for an arbitrary number of spatial dimensions and a variety of coordinate systems. The main design is for linear $\mathcal{D}$, but nonlinear operators can also be used.

The code has been used primarily for solving the time-dependent Schrödinger equation of atomic and molecular systems in ultrashort pulses and strong near-IR fields. It employs three newly developed key techniques: irECS — *"infinite range Exterior Complex Scaling"* [35], tSurff — the *time-dependent Surface flux* method [36] for photo-emission spectra, and hacc — the *"hybrid anti-symmetrized coupled channels"* method [23]. Two of these are reflected in the code's name tRecX = tSurff + irECS.

The code is open-source hosted on a Gitlab repository [14]. It is written mostly in C++ and linked with some Fortran-based libraries. Compilation is through CMake, and Doxygen documentation is available. Tutorials and further materials are available in the git repository and from the tRecX website [37].

2.5 XChem

XChem [1, 24, 25] is a solution for an all-electron *ab initio* calculation of the electronic continuum of molecular systems that combines, by means of a hybrid Gaussian + B-spline basis [24], existing quantum-chemistry packages for bound-state calculations (as implemented, e.g., in MOLCAS [2]) with close-coupling (CC) scattering methods to describe the correlated electronic continuum wave function. In particular, XChem permits the description of electron correlation in the molecular electronic continuum at a level of accuracy similar to that reached by equivalent configuration interaction methods for bound-state calculations, so that autoionization, shake-up, and inter-channel couplings can be accurately described. The method is appropriate to describe the single-ionization continuum of atoms [25, 26], as well as small and medium-size molecules [22, 27].

The code yields the scattering states of the molecular system via the eigenstates of the close-coupling matrix. From the calculated scattering states, XChem provides total and partial photoionization cross sections, resonance positions, and the corresponding autoionization widths. An additional, very useful feature of XChem is that it provides all the dipole coupling matrix elements (bound-bound, bound-continuum, and continuum-continuum) required to solve the time-dependent Schrödinger equation (TDSE) that describes the interaction of molecules with ultrashort pulses. This allows one, for instance, to simulate attosecond experiments in which combinations of several pulses lead to complex interferometric patterns in the ionization continuum [17].

2.6 ePolyScat

A new addition to the AMPGateway is ePolyScat. ePolyScat is a code that computes electron-molecule scattering and molecular photoionization cross sections [21, 29] within the fixed-nuclei approximation. Using the adiabatic approximation, it is possible to extract ro-vibrationally resolved cross sections. The code is limited to initial and target states that are well-represented by a single configuration state function, and the final continuum state is limited to a single electronic state. The interaction potential between the electron and the molecule is then given by the static-exchange potential, which can be modified by approximate functionals to describe target polarization effects. The calculations are performed in the full symmetry point group for the molecular geometry, including both abelian and many non-abelian point groups.

The external source of bound molecular orbitals required for ePolyScat may be obtained from molden files produced by many standard quantum-chemistry codes. The continuum wave functions are obtained as single-center expansions using products of symmetry-adapted angular harmonics and radial functions represented on a grid of points. The size of the single-center expansion is controlled by a maximum angular momentum parameter, and the density of the radial grid is controlled by a maximum energy parameter. The code is written in Fortran2003 and uses MPI and LAPACK libraries. The current version can efficiently use up to about 32 processes but requires at least two.

2.7 MESA

We decided to also include the rather old Many Electron Structure Applications (MESA) [32] code that was developed at Los Alamos in the 1980s and modified to compute electron scattering and photoionization cross sections from polyatomic molecules using the Complex Kohn Method [31]. While this code is old, poorly documented, and in need of significant modernization, it was built to compute electron scattering and photoionization transition matrix elements for general polyatomic molecules, a capability not present in other codes and of interest to many users. The code can include electron correlation using an *ab-initio* optical potential and can handle many electronic channels.

Code	Application	Method	Parallel or Serial	Access	Limitations
BSR	Electron scattering, photoionization, structure	R-matrix/B-spline	S, MPI	Public (Git)	Atoms, atomic ions
CCC	Electron scattering, Positron scattering Photoionization	Close-coupling, Fredholm equations in momentum space	OpenMP and MPI	Public (Git)	Quasi one- and two-electron atoms and ions
UKRMol+	Electron scattering, photoionization	R-matrix, Close-, coupling, Gaussian and B-spline basis	OpenMP and MPI	Public (Zenodo)	Molecules and clusters ($\lesssim$ 30 atoms)
tRecX	Strong-field photo-emission spectra, rates	TDSE (tSurff, haCC): grids, CI-states, FE-DVR bases	MPI	Public (Git)	Small molecules, two-electron atoms
XChem	Scattering states Photoionization	Close-coupling, CI, Hybrid Gaussian & B-spline basis	OpenMP	By request	Small and medium-size molecules
MESA	Electronic structure, scattering, photoionization	SCF, MCSCF, CI, Optical potential, Complex Kohn	S	By request	Small to medium-size molecules
ePolyScat	Electron scattering, photoionization	Single-center expansion, one electronic channel	MPI	By request	Small to medium-size molecules

Table 1: Some characteristics of the software suites deployed in the AMPGateway.

3 AMP SCIENCE GATEWAY: APPLICATION DEPLOYMENT AND GATEWAY INTEGRATION

The science gateway is a portal that requires hosting a web site and deploying services such as a content management system to provide a gateway theme and documentation, user authentication and authorization, science application registration, user interface development, and file and other data management. The AMPGateway, ampgateway.org, is deployed using the Apache Airavata framework. [30]. The Science Gateway Platform as a Service (SciGaP), an NSF-supported project that serves multi-tenanted Apache Airavata middleware hosting services [30] at Indiana University, provides the servers and primary storage for the gateway. The SciGaP platform delivers all the services needed for the production gateways, such as gateway registration and maintenance, HPC registry, and authentication on high-availability and redundant hardware resources maintained by Indiana University IT services. It also enables the Airavata functions, such as workflow orchestration, log and email mining for distributed message brokering, and data management. The Airavata core provides data models and APIs that enable gateway features such as managing user identity, accounts, authorization and sharing provisioning, the ability to access XSEDE and other high-performance computational resources, task creation for scientific workflows, workflow information management, and user interfaces for data for the workflows. The SciGaP registered HPC resources are transparently integrated into the AMPGateway, and the batch schedulers are used for queuing the execution of workflows consisting of applications and user-defined parameters/data and their dependencies. It also provides monitoring and status updates, as well as access to the output data and results in raw and post-processed forms.

3.1 Django-based Gateway Frontend

We deployed a Django web framework based frontend for the AMPGateway to provide the user-facing environment and integrated the Airavata through a python API for the gateway functionality. The Python and Vue JS Javascript based Django framework provides significant flexibility in molding the user interfaces and integration of application specific pre- and post-processing tools. The previously deployed PHP based reference gateway is now deprecated.

3.1.1 Gateway Theme Creation. The gateway theme is important as it provides immediate and direct interaction with the stakeholders, funding agencies, peer scientists, developers, users, and trainees. It must provide an attractive and easy to navigate portal to support the requirements of all these stakeholders. Funding agencies are perpetually interested in the impact of the project on the scientific community, while the peer scientists are typically interested in issues of use or how they can contribute. Users and trainees require documentation and training materials. The AMPGateway integrates the Wagtail content management system (CMS) to drive the theme and enable gateway managers to edit, review, and preview the look and feel of the portal, save drafts, and eventually publish the pages for public consumption. Some functions of the CMS are depicted in Fig. 1.

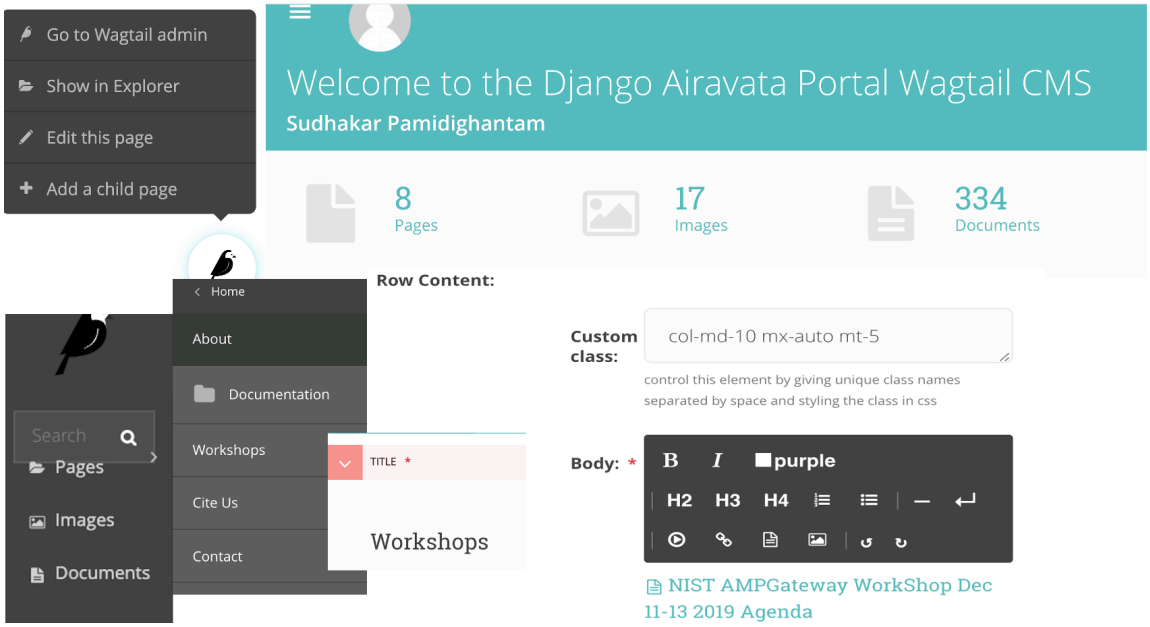

Figure 1: Wagtail content management system to edit AMPGateway web pages showing a "Workshop" tab creation with a hyperlink addition.

3.2 User Access and Control

The AMPGateway user access is managed by the Keycloak [9] identity management system supported by SciGaP. Access can be created by users, who set a userid and password pair and provide an email address for verification purposes. In addition, an automated registration process using the user institutional login via CI-Logon [4] is also available and avoids the email verification step. An advanced user authorization at a computational resource through local LDAP subtree modification has also been prototyped for preauthorized users on that resource. The gateway administrator controls access to the resources and can set up a verification and approval model for a user to access gateway resources. The administrators and users can create groups and add specific collaborators to their group to facilitate access to resources (computational and application software) and data (experiment workflows, files, images, etc). AMPGateway administrators grant approval by updating the user group to a default "Gateway User" group to provide a basic access. A "gateway-user" can then use gateway services such as creating, monitoring, sharing, and cloning experiments (computational simulations). The users can also add their own compute resource allocations using the functions available under the "User Settings" profile. Gateway Admin group users will have additional privileges to modify the gateway themes, application integration, and user interface creation. The admins also have privileges to examine gateway-wide usage, monitor failures, and dive into the causes thereof to pro- or retro-actively take actions on the gateway. They can use these data for sharing with supporting stakeholders or for reporting purposes. The "admin" users also have authorization to control metadata for accessing XSEDE resources through the gateway "community login," manage users, and monitor and access all user experiments. These privileges enable admins to troubleshoot any issues relating to user services efficiently. Multiple admin users can be enabled by adding any user to an "Admin" group. Users can also create their own groups and add collaborators to the group for sharing computations and/or data. Potential use cases include group organization of students in a class or participants in a project for sharing class-wide or project-wide, respectively.

3.3 AMP Application Deployment and User Interface Creation

The AMP gateway started with five specific applications described above: the BSR, CCC, UKRMol+, tRecX, and Xchem suites. These applications were compiled and tested on a number of XSEDE resources. Each one requires a different set of libraries and, in the case of XChem, integration with other open-source software such as OpenMolcas [3]. Furthermore, UKRMol+ needs the PSI4 [38] quantum-chemistry application. The suites were independently tested by the collaborating XSEDE ECSS consultant when deployed by the developers or deployed by the consultant to establish the required environments and to tweak make/cmake data. Additionally, the application software can be wrapped by Shell, Perl, or Python scripts to enhance input processing and execution preparation. The application integration into the AMPGateway consists of three steps: 1. Defining the application as a "module" to identify the application; 2. Defining an "interface" for user interaction with the application; and 3. Registering a resource-specific "deployment" description to define it fully on the gateway. The user interfaces are tailored to each application and enable the users to provide input parameters using files or variables that either can be sent as arguments to the application or via a wrapping script. Currently, the interface generator provides ways to define file URIs, variables (strings,

real/integer/Booleans) for inputs and standard error/outputs by default, and file and variables for outputs. The Django input fields can be extended to include conditional inputs, cascading menu-driven inputs, radio button/check boxes, drop-down menu-driven options, and, in advanced cases, graphical application plugins.

The job submission interface then enables users to define HPC resource details such as the system/machine ID that is automatically sorted from the list obtained from the deployment description, queue/partition and allocation registered by the administrator or a user, and job specific details such as the number of nodes and processors, run time, and memory specifications. In addition to these experiment specifications, the Apache Airavata framework provides a sharing mechanism for jobs/experiments and projects (which are collections of experiments) to be shared with individual collaborators or groups. The sharing choice can be set during the experiment creation itself, or at any time after. This allows for collaborative job submission, monitoring and analysis of the results, and training new users quickly.

Figure 2 provides the input field enhancement using JSON-style metadata to provide a dynamic user interface for the tRecX application to ingest an optional JobID to reuse data from a previous run, continue the simulation, and potentially collect data for comparative analysis, and post-processing (see next section).

3.4 Monitoring Job Progress

During experiment execution, messages are sent to a message bus for monitoring job status. These messages are used to update the status on the gateway. Additionally, once the experiment is accepted by the gateway and launched, an "Experiment Summary" interface is created and automatically refreshed periodically to show the status of the job submitted into the XSEDE resources. The Experiment Summary page will provide links to application outputs, scheduler error and standard outputs, links to Gateway storage for the experiment, as well as the post-processed output in the form of visualizations. The post-processing functions can be defined independently as a plugin application, and invoked for certain kinds of data such as plot data. Multiple mime types for a given output file can be supported and data can be presented in raw or post-processed form as a graph/movie as registered in the output field configuration data model. These mime types are used by the browser to render the data within the experiment summary page itself.

3.5 Gateway Administrator Dashboard

The "Admin Dashboard" is the working space for the gateway administrator (admin) within the gateway. All the admin features such as application management, user management, HPC integration, group and security management are available through the Admin Dashboard. This dashboard provides interfaces to manage security credentials and a way to set and secure communications to gateway compute and storage resources. The gateway admin can set compute resources and define preferences for the usage at each of the compute resources with specifications such as community or shared login name, scratch location for the job data staging and execution, preferred job submission and file transfer protocols, allocation project number for charging the run time, and other scheduler choices. Similarly, the admin dashboard is also used to generate an SSH credential for secure authentication and authorization at the compute and storage resource communications.

The admins can also monitor all gateway experiments using the "Experiment Statistics" function available in the "Admin Dashboard." The monitoring of experiment information is provided by a log processing system that extracts the relevant task level execution logs from the gateway middleware and presents it in the gateway monitoring interfaces, as depicted in Fig. 3. The "Experiment Statistics" function can be used to obtain experiments sorted by date, user, application software, resource, and combination-specific constraints. For a given experiment, a detailed task-level description of the experiment workflow can be obtained. This summary also provides job scheduler scripts created by the Airavata middleware, the actual path of the job directory on the remote HPC system, and detailed task-level logs. The admin can use this to check the health of the job workflow or to troubleshoot, if needed. The admin dashboard provides a feature that creates notifications to all users to communicate important events such as news on application deployments, resource down times, changes in policy, etc.

4 COMMUNITY PARTICIPATION AND OUTLOOK

We recently conducted a workshop [11] funded in part by NIST, the MOLecular Software Sciences Institute [12], and the NSF to invite the community to experience the gateway firsthand, provide guidance for future enhancements, suggest additions and changes, and explore mechanisms for sustaining support for the future. The workshop consisted of presentations on the science underlying the codes and demonstrations of how they are deployed on the gateway to solve specific, representative problems chosen by the presenter. The last half day of the workshop was devoted to fleshing out a prioritized list of what was needed to make the AMPGateway most useful to the developers as well as to a broader community. One immediate result of the workshop was to deploy a new application, ePolyScat, developed by Lucchese, into the gateway. The ePolyScat application is now deployed on Stampede2 and made available through the gateway. We are currently deploying the LBL NERSC Cori HPC resource for community/collaborative computing.

Some of the more critical issues that emerged from the discussion included a prioritized list for further development:

- **Usability:** Editing of inputs on the gateway, a gateway-wide framework for GUIs, methods for wrapping complex work flows, integration multi-component software suites into a single module.
- **Storage management:** Providing standard storage formats and a hierarchy of scratch/volatile/ permanent file storage. At present, data reside on the production machines in native code format.
- **Visual and functional design:** Setting up a developmental web server for gateway development by code deployers.
- **Ticket and community discussion:** Setting up a ticketing system for gateway-related questions; provide a forum for code- and science-specific questions.
- **Dissemination:** Preparing a journal article with examples of code-usage codes on the gateway.

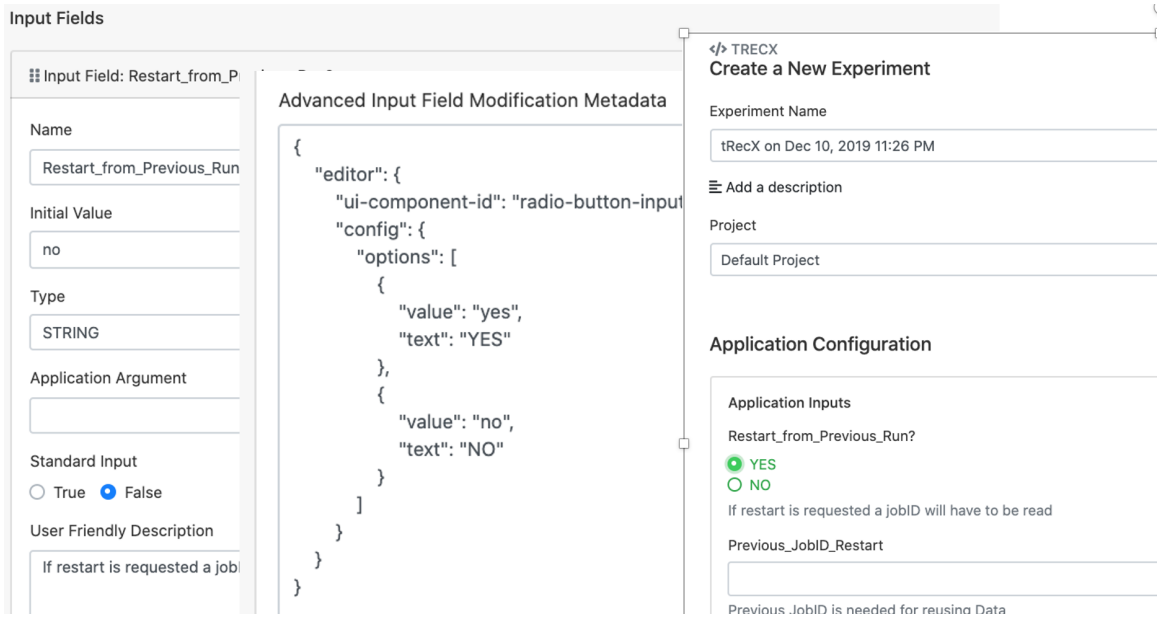

Figure 2: Dynamic input interface for tRecX software to enable a restart from a previous run based on a jobID. The two left-most panels show the administrator-edited interface with code and metadata input fields. Users see the right panel as the finished interface.

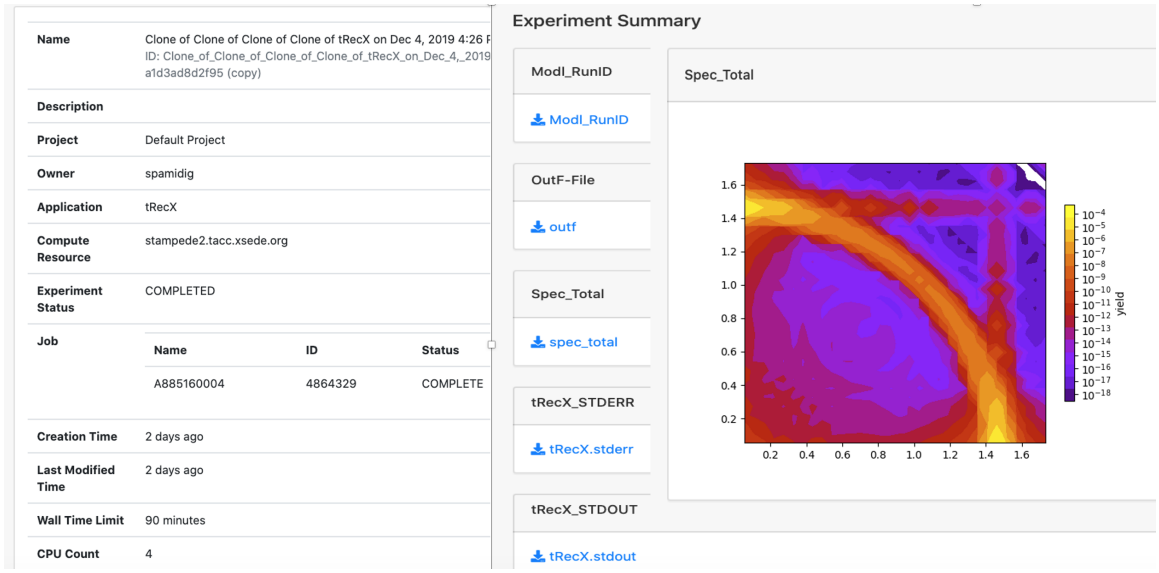

Figure 3: Experiment summary page for a tRecX job with the output choice to plot using a Python plugin integrated in the AMPGateway.

- **Benchmarks and comparisons:** Seting up mechanisms for data-exchange and comparisons between alternative codes; providing benchmark results where possible.

ACKNOWLEDGMENTS

BIS acknowledges the Mathematical Software group of the Applied and Computational Mathematics Division at NIST for supporting this work. This work used the Extreme Science and Engineering Discovery Environment (XSEDE) Stampede2, Comet, and Bridges resources, and an ECSS collaboration grant through the allocation TG-PHY180023, which is supported by National Science Foundation under grant No. ACI-1548562. Additional support is provided by the Science Gateways Community Institute, NSF Award No. 1547611 [18]. The SciGaP.org platform and Apache Airavata are supported

by NSF Award No.1339774. KB, OZ, and KRH acknowledge NSF support through grant Nos. PHY-1520970, PHY-1803844, and OAC-1834740. FM and JG-V acknowledge support by the ERC Proof-of-Concept Grant No. 780284, and by the Spanish MINECO Grant No. FIS2016-77889R. JT and JDG acknowledge support through the UK-AMOR high-end computing consortium under EPSRC grant EP/R029342/1. IB is grateful for the support of the Australian Research Council. Work performed at LBNL was supported by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, Chemical Sciences, Geosciences, and Biosciences Division under Contract No. DE-AC02-05CH11231.

REFERENCES

[1] Luca Argenti, Inés Corral, Jesús González-Vázquez, Markus Klinker, Carlos Marante, and Fernando Martín. 2017. https://www.xchem.uam.es/xchem/?tag=xchem

[2] The Molcas Authors. 2016. Molcas 8: New capabilities for multiconfigurational quantum chemical calculations across the periodic table. *Journal of Computational Chemistry* 37, 5 (2016), 506–541. https://doi.org/10.1002/jcc.24221

[3] The OpenMolcas Authors. 2019. https://gitlab.com/Molcas/OpenMolcas

[4] Jim Basney, Heather Flanagan, Terry Fleury, Jeff Gaynor, Scott Koranda, and Benn Oshrin. 2019. CILogon: Enabling Federated Identity and Access Management for Scientific Collaborations. In *International Symposium on Grids & Clouds 2019 (ISGC2019) - Network, Security, Infrastructure & Operations*, Vol. 351. https://doi.org/10.22323/1.351.0031

[5] Igor Bray. 1994. Convergent close-coupling method for the calculation of electron scattering on hydrogenlike targets. *Phys. Rev. A* 49, 2 (1994), 1066–1082. https://doi.org/10.1103/PhysRevA.49.1066

[6] Igor Bray and Andris T. Stelbovics. 1992. Convergent close-coupling calculations of electron-hydrogen scattering. *Phys. Rev. A* 46, 11 (1992), 6995–7011. https://doi.org/10.1103/PhysRevA.46.6995

[7] Igor Bray and Andris T. Stelbovics. 1995. Calculation of Electron Scattering on Hydrogenic Targets. In *Advances In Atomic, Molecular, and Optical Physics*. Vol. 35. 209 – 254. https://doi.org/10.1016/S1049-250X(08)60164-0

[8] Philip G. Burke. 2011. *R-Matrix Theory of Atomic Collisions*. Springer Series on Atomic, Optical, and Plasma Physics, Vol. 61. https://doi.org/10.1007/978-3-642-15931-2

[9] Marcus Christie, Anuj Bhandar, Supun Nakandala, Suresh Marru, Eroma Abeysinghe, Sudhakar Pamidighantam, and Marlon Pierce. 2017. Using Keycloak for Gateway Authentication and Authorization. https://doi.org/10.6084/m9.figshare.5483557.v1

[10] The AMP Gateway Community. 2019. https://ampgateway.org/

[11] The AMP Gateway Community. 2019. https://ampgateway.org/workshops/

[12] Daniel T. Crawford, Cecilia Clementi, Robert Harrison, Teresa Head-Gordon, Shantenu Jha, Anna Krylov, and Theresa Windus. 2016. https://molssi.org/

[13] Andrew C. Brown *et al.* 2020. RMT: R-matrix with time-dependence. Solving the semi-relativistic, time-dependent Schrödinger equation for general, multi-electron atoms and molecules in intense, ultrashort, arbitrarily polarized laser pulses. *Comput. Phys. Commun.* (2020). https://doi.org/10.1016/j.cpc.2019.107062

[14] Armin Scrinzi *et al.* 2016. https://gitlab.physik.uni-muenchen.de/AG-Scrinzi

[15] Joanne M. Carr *et al.* 2012. UKRmol: a low-energy electron- and positron-molecule scattering suite. *The European Physical Journal D* 66, 3 (2012), 58. https://doi.org/10.1140/epjd/e2011-20653-6

[16] John Towns *et al.* 2014. XSEDE: Accelerating Scientific Discovery. *Computing in Science Engineering* 16, 5 (2014), 62–74. https://doi.org/10.1109/MCSE.2014.80

[17] Lou Barreau *et al.* 2019. Disentangling Spectral Phases of Interfering Autoionizing States from Attosecond Interferometric Measurements. *Phys. Rev. Lett.* 122, 25 (2019), 253203. https://doi.org/10.1103/PhysRevLett.122.253203

[18] Marlon E. Pierce *et al.* 2015. Apache Airavata: design and directions of a science gateway framework. *Concurrency and Computation: Practice and Experience* 27, 16 (2015), 4282–4291. https://doi.org/10.1002/cpe.3534

[19] Dmitry V. Fursa and Igor Bray. 1995. Calculation of electron-helium scattering. *Phys. Rev. A* 52, 2 (1995), 1279–1297. https://doi.org/10.1103/PhysRevA.52.1279

[20] Dmitry V. Fursa and Igor Bray. 1997. Convergent close-coupling calculations of electron scattering on helium-like atoms and ions: electron - beryllium scattering. *Journal of Physics B: Atomic, Molecular and Optical Physics* 30, 24 (1997), 5895–5913. https://doi.org/10.1088/0953-4075/30/24/023

[21] Franco A. Gianturco, Robert R. Lucchese, and Nico Sanna. 1994. Calculation of low-energy elastic cross sections for electron-CF_4 scattering. *The Journal of Chemical Physics* 100, 9 (1994), 6464–6471. https://doi.org/10.1063/1.467237

[22] Markus Klinker, Carlos Marante, Luca Argenti, Jesús González-Vázquez, and Fernando Martín. 2018. Electron Correlation in the Ionization Continuum of Molecules: Photoionization of N_2 in the Vicinity of the Hopfield Series of Autoionizing States. *The Journal of Physical Chemistry Letters* 9, 4 (2018), 756–762. https://doi.org/10.1021/acs.jpclett.7b03220

[23] Vinay P. Majety, Alejandro Zielinski, and Armin Scrinzi. 2015. Photoionization of few electron systems: a hybrid coupled channels approach. *New Journal of Physics* 17, 6 (2015), 063002. https://doi.org/10.1088/1367-2630/17/6/063002

[24] Carlos Marante, Luca Argenti, and Fernando Martín. 2014. Hybrid Gaussian–B-spline basis for the electronic continuum: Photoionization of atomic hydrogen. *Phys. Rev. A* 90, 1 (2014), 012506. https://doi.org/10.1103/PhysRevA.90.012506

[25] Carlos Marante, Markus Klinker, Inés Corral, Jesús González-Vázquez, Luca Argenti, and Fernando Martín. 2017. Hybrid-Basis Close-Coupling Interface to Quantum Chemistry Packages for the Treatment of Ionization Problems. *Journal of Chemical Theory and Computation* 13, 2 (2017), 499–514. https://doi.org/10.1021/acs.jctc.6b00907

[26] Carlos Marante, Markus Klinker, Tor Kjellsson, Eva Lindroth, Jesús González-Vázquez, Luca Argenti, and Fernando Martín. 2017. Photoionization using the xchem approach: Total and partial cross sections of Ne and resonance parameters above the $2s^2 2p^5$ threshold. *Phys. Rev. A* 96, 2 (2017), 022507. https://doi.org/10.1103/PhysRevA.96.022507

[27] Sonia Marggi Poullain, Markus Klinker, Jesús González-Vázquez, and Fernando Martín. 2019. Resonant photoionization of O_2 up to the fourth ionization threshold. *Phys. Chem. Chem. Phys.* 21, 30 (2019), 16497–16504. https://doi.org/10.1039/C9CP02150G

[28] Zdeněk Mašín, Jakub Benda, Jimena D. Gorfinkiel, Alex G. Harvey, and Jonathan Tennyson. 2020. UKRmol+: A suite for modelling electronic processes in molecules interacting with electrons, positrons and photons using the R-matrix method. *Comput. Phys. Commun.* 249 (2020), 107092. https://doi.org/10.1016/j.cpc.2019.107092

[29] Alexandra P. P. Natalense and Robert R. Lucchese. 1999. Cross section and asymmetry parameter calculation for sulfur 1s photoionization of SF6. *The Journal of Chemical Physics* 111, 12 (1999), 5344–5348. https://doi.org/10.1063/1.479794

[30] Marlon. E. Pierce, Suresh Marru, Eroma Abeysinghe, Sudhakar Pamidighantam, Marcus Christie, and Dimuthu Wannipurage. 2018. Supporting Science Gateways Using Apache Airavata and SciGaP Services. In *PEARC '18: Proceedings of the Practice and Experience on Advanced Research Computing*.

[31] Thomas N. Rescigno, C. William McCurdy, Ann E. Orel, and Byron. H. Lengsfield III. 1995. *The Complex Kohn Variational Method*. https://doi.org/10.1007/978-1-4757-9797-8_1

[32] Paul W. Saxe, Byron H. Lengsfield III, Richard L. Martin, Miles Page, Barry I. Schneider, and P. Jeffrey Hay. 1985. The MESA code was developed at LANL as a pure quantum chemistry code. A number of features were added to include those parts of the molecular short-range continuum that could be described using Gaussian basis sets.

[33] Barry I. Schneider, Klaus Bartschat, Oleg Zatsarinny, Igor Bray, Armin Scrinzi, Fernando Martín, Markus Klinker, Jonathan Tennyson, Jimena D. Gorfinkiel, and Sudhakar Pamidighantam. 2020. A Science Gateway for Atomic and Molecular Physics. arXiv:2001.02286

[34] Barry I. Schneider, Robert Forrey, and Naduvalath Balakrishnan. 2018. https://www.cfa.harvard.edu/itamp-event/developing-flexible-and-robust.-software-computational-atomic-and-molecular-physics-0

[35] Armin Scrinzi. 2010. Infinite-range exterior complex scaling as a perfect absorber in time-dependent problems. *Phys. Rev. A* 81, 5 (2010), 053845. https://doi.org/10.1103/PhysRevA.81.053845

[36] Armin Scrinzi. 2012. t-SURFF: fully differential two-electron photo-emission spectra. *New Journal of Physics* 14, 8 (2012), 085008. https://doi.org/10.1088/1367-2630/14/8/085008

[37] Armin Scrinzi. 2016. https://trecx.physik.uni-muenchen.de/index.html

[38] C. David Sherrill, T. Daniel Crawford, Francesco Evangelista, Henry F. Schaefer III, Justin Turney, Andy Simmonett, Rollin King, and Lori Burns. 2009. http://www.psicode.org

[39] Jonathan Tennyson. 2010. Electron - molecule collision calculations using the R-matrix method. *Phys. Rep.* 491 (2010), 29–76.

[40] Oleg Zatsarinny. 2006. BSR: B-spline atomic R-matrix codes. *Computer Physics Communications* 174, 4 (2006), 273 – 356. https://doi.org/10.1016/j.cpc.2005.10.006

[41] Oleg Zatsarinny and Klaus Bartschat. 2013. The B-spline R-matrix method for atomic processes: application to atomic structure, electron collisions and photoionization. *Journal of Physics B: Atomic, Molecular and Optical Physics* 46, 11 (2013), 112001. https://doi.org/10.1088/0953-4075/46/11/112001

[42] Zenodo. 2020. https://zenodo.org