The C-MĀIKI Gateway: A Modern Science Platform for Analyzing Microbiome Data

Sean B. Cleveland
University of Hawaii - System
Honolulu, HI, USA
seanbc@hawaii.edu

Cédric Arisdakessian
University of Hawaii - Manoa
Honolulu, HI, USA
carisdak@hawaii.edu

Craig E. Nelson
University of Hawaii - Manoa
Honolulu, HI, USA
cen@hawaii.edu

Mahdi Belcaid
University of Hawaii - Manoa
Honolulu, HI, USA
mbelcaid@hawaii.edu

Kiana L. Frank
University of Hawaii - Manoa
Honolulu, HI, USA
klfrank@hawaii.edu

Gwen A. Jacobs
University of Hawaii - System
Honolulu, HI, USA
gwenjh@hawaii.edu

ABSTRACT

In collaboration with the Center for Microbiome Analysis through Island Knowledge and Investigations (C-MĀIKI), the Hawaii EPSCoR Ike Wai project and the Hawaii Data Science Institute, a new science gateway, the C-MĀIKI gateway, was developed to support modern, interoperable and scalable microbiome data analysis. This gateway provides a web-based interface for accessing high-performance computing resources and storage to enable and support reproducible microbiome data analysis. The C-MĀIKI gateway is accelerating the analysis of microbiome data for Hawaii through ease of use and centralized infrastructure.

CCS CONCEPTS

• **Information systems → Computing platforms**.

KEYWORDS

datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
Sean B. Cleveland, Cédric Arisdakessian, Craig E. Nelson, Mahdi Belcaid, Kiana L. Frank, and Gwen A. Jacobs. 2022. The C-MĀIKI Gateway: A Modern Science Platform for Analyzing Microbiome Data. In *Practice and Experience in Advanced Research Computing (PEARC '22), July 10–14, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3491418.3530291

1 INTRODUCTION

The Microbiome is composed of microbes which include bacteria, fungi and viruses and are an integral part of the biosphere. Microbes are relied on by animals and plants extensively for carrying out many critical processes that span fighting off disease to sustaining the air we breathe and the water we drink. In fact, microbes occupy almost every surface of the human body and are the most abundant organisms on Earth, with a diversity estimated at close to 1 trillion species [17]. While the last decade has produced a surge in microbiome research, the field is still considered in its infancy. This research commonly produces large datasets from next generation sequencing of microbiomes which can be challenging to manage. The analysis of these datasets often requires advanced computational tools and workflows in addition to familiarity with the complex cyberinfrastructures required to run the analyses at scale.

Currently there are a number of tools and pipelines for analyzing the most widespread and fundamental types of microbiome data which includes taxonomic abundance profiles from high throughput DNA amplicon sequencing (e.g. amplicon surveys of marker genes such as 16S, ITS or CO1). Notable among these tools are Mothur[20], QIIME2[7] metaAMP[12], VSEARCH/USEARCH[19] and DADA2[10]. These tools provide an extensive ecosystem of subprograms to carry out the vast majority of the steps involved in the analysis of microbiome marker data. Often, rather than selecting a single one of these tools or pipelines, experienced researcher will implement a customized workflows choosing subprograms from each of these ecosystems to fit their exact needs. Unfortunately, the lack of smooth interoperability across these ecosystems can make designing and implementing these hybrid pipelines difficult due to connecting inputs and outputs of subprograms that may be in custom or differing formats. Further, extensive dependencies associated with such pipelines render their deployment time-consuming for experts and impossible for novices, particularly when deploying to new computational environments. Similarly, updating pipelines to account for new subprograms often involves writing new code or scripts, making them challenging to maintain and update. With the increasing popularity of microbiome research coupled and a flood of new bioinformatics, computational tools and the rapid increases in data volume accompanying new sequencing technology, there is a large need for cyberinfrastructure solutions to enhance usability, interoperability and scalability for microbiome analysis.

To overcome some of these challenges a collaboration between the Center for Microbiome Analysis through Island Knowledge and Investigations (C-MAIKI), the Hawaii EPSCoR Ike Wai project and the Hawaii Data Science Institute developed a set of pipelines - MetaFlow|mics[2] - for automating the processing of microbiome data. This suite of tools, although relatively straightforward to use, still requires knowledge of the underlying computational resource to deploy and experience with command line

PEARC '22, July 10–14, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9161-0/22/07…$15.00
https://doi.org/10.1145/3491418.3530291

tools which can be obstacles towards adoption. In order to promote MetaFlow|mics adoption, enhance usability and simplify the overall workflow we developed and deployed the C-MĀIKI Science Gateway (https://cmaiki.its.hawaii.edu].

The C-MĀIKI gateway provides a web-based interface for accessing advanced high-performance computing resources and storage to support and accelerate microbiome research. By leveraging the Tapis framework, this gateway makes microbial sequence analysis workflows and data easier to access, launch, track, manage and reproduce through a user-friendly web-accessible interface. The extendable application interface currently supports the MetaFlow|mics pipeline analyses and provides the tracking of all input/output data and analysis parameters to provide notifications and provenance information for every analysis run.

In this paper we will describe the C-MĀIKI gateway features, implementation, user interfaces and future opportunities.

2 BACKGROUND

2.1 Science Gateways

Science gateways, virtual laboratories and virtual research environments are all terms used to refer to community-developed digital environments that are designed to meet a set of needs for a research community[22][4]. These Science gateways are frequently implemented as Web and mobile applications, providing access to community resources such as software, data, collaboration tools, instrumentation, and high-performance/cloud computing[16]. These research environments are enabling significant contributions to many research domains, facilitating more efficient, open, reproducible research in ways that accelerate science.

2.2 Tapis

Tapis is an open source, NSF funded Application Program Interface (API) platform for distributed computation. It provides production-grade capabilities to enable researchers to 1) securely execute workflows that span geographically distributed providers, 2) store and retrieve streaming/sensor data for real-time and batch job processing with support for temporal and spatial indexes and queries, 3) leverage containerized codes to enable portability, and reduce the overall time-to-solution by utilizing data locality and other "smart scheduling" techniques, 4) improve repeatability and reproducibility of computations with history and provenance tracking built into the API and 5) manage access to data and results through a fine-grained permissions model, so that digital assets can be securely shared with colleagues or the community at large. Researchers and applications are able to interact with Tapis by making authenticated HTTP requests to Tapis's public endpoints. In response to requests, Tapis's network of microservices interact with a vast array of physical resources on behalf of users including high performance and high throughput computing clusters file servers and other storage systems, databases, bare metal, and virtual servers. Tapis aims to be the underlying cyberinfrastructure for a diverse set of research projects: from large scale science gateways built to serve entire communities, to smaller projects and individual labs wanting to automate one or more components of their process.

Tapis can be leveraged as a hosted solution or distributed between various institutions. Tapis is multi-tenant, meaning that there can be a number of organizations (i.e. a grouping of users, such as an institution, lab or project) using the same set of Tapis API services but persisting data in logically separate, secure, namespace. The central hosted instance is currently hosted by the Texas Advanced Computing Center (TACC) at the University of Texas at Austin. Other institutions, such as the University of Hawaii (UH), have a hybrid deployment with subsets of Tapis API services deployed locally while leveraging others hosted at TACC. The UH Tapis instance is what the C-MĀIKI gateway leverages for backend API services.

2.3 MetaFlow|mics

MetaFlow|mics is a collection of three pipelines that address analysis for some of the most popular metagenomic marker-genes (Figure 1). The first pipeline is a "demultiplexing" tool that assigns sequences to their corresponding samples in pooled-sample sequencing runs based on oligonucleotide indexes [14][15][8][21][9]. The second and third are end-to-end analysis pipelines specifically designed for bacterial(16s) and fungal (ITS) marker gene amplicon metagenomics, respectively. Both of these pipelines include the most common analysis steps:

(1) Applying quality control filters, instrument "denoising" algorithms[10][18] and assemble paired-end reads into contiguous sequences.
(2) Identifying and purging rare, contaminated and/or chimeric sequences.
(3) Clustering the final sequences based on their sequence similarity to define Operational Taxonomic Units (OTUs), which are proxies for microbial species or strain.
(4) Annotating OTUs with their respective taxonomic lineage.
(5) Computing various ecological and evolutionary metrics, such as alpha and beta diversity and phylogenetic relatedness.

All of these steps are implemented using R, python, or a dedicated compiled algorithm (e.g. VSEARCH or Mothur).

3 USAGE AND IMPACT

To date the C-MĀIKI gateway has enabled more than 44 researchers, graduates and undergraduates to run over 1,250 pipeline jobs which breaks out into more than 805,000 parallel sub-jobs. These computational jobs have enabled users to access more than 150,000 CPU hours and process more than 6 TB of data. If these sub-jobs would have run sequentially it would have taken close to 17 years to process the data up to this point.

4 IMPLEMENTATION AND FEATURES

The C-MĀIKI gateway was designed and implemented with the objectives of promoting the adoption of MetaFlow|mics, facilitating analysis of microbiome data and maximizing usability.

4.1 Authentication and Authorization

To support authentication the C-MĀIKI gateway integrates the Tapis identity services with the UH LDAP service for enabling UH credential usage and leverages the UH affiliate account provisioning process for external collaborators in addition to a gateway LDAP

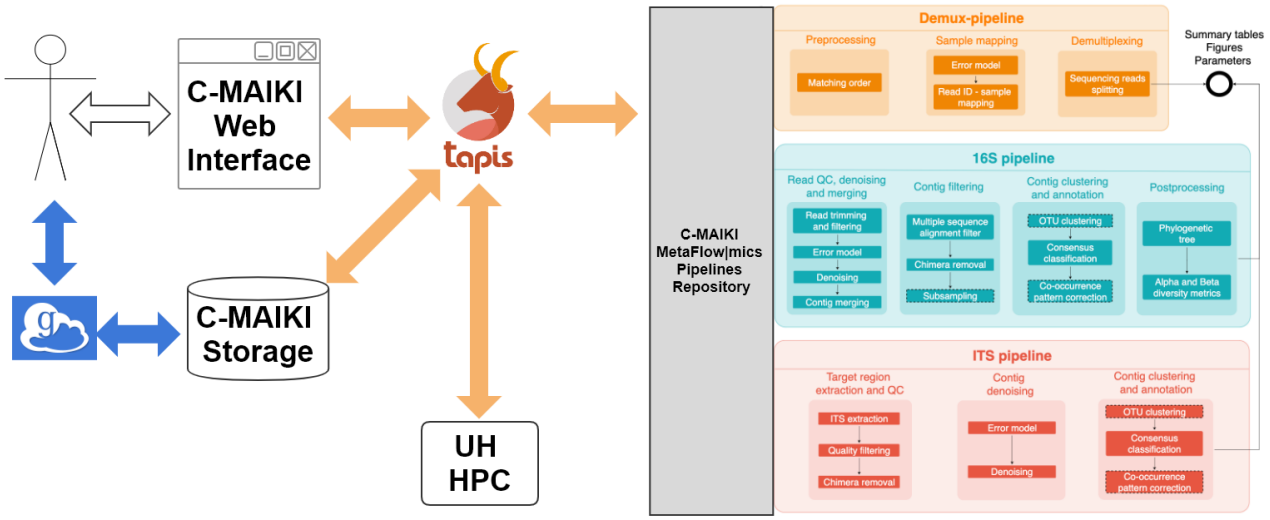

Figure 1: Diagram of the C-MĀIKI gateway and how Tapis(orange arrows) connects the user interface, storage, compute and the MetaFlow|mics software pipelines (Demultiplexing pipeline, 16S pipeline, and ITS pipeline). Globus (blue) also provide a second interface for storage access.

for accounts that fall outside those user, such as third party application accounts and service accounts. The usage of Tapis allows the gateway to support OAuth flows but primarily relies on the password flow where a token and refresh token are generated for a set length of validity (typically four hours). Authorization is provided by Tapis Tokens API and data and metadata authorization is handled by the Tapis APIs which allow Access Control Levels (ACLs) on individual files and metadata objects in addition to the gateway storage and compute resources.

4.2 User Interface

The user interface has been separated from the backend implementation by leveraging Tapis as the middleware to power the backend services (Figure 2). This separation allows continued update/maintenance of the backend by a wider development community, the Tapis community, and the UI to be focused towards the microbiome community in the C-MĀIKI gateway.

To provide cross-platform usability, the C-MĀIKI gateway user interface (UI) must function across desktop web environments and mobile devices, especially for the wider community. The UI for the gateway is a fully responsive, mobile friendly javascript web application. The gateway UI is a based on Agave ToGo[13] and has been customized to match the needs and workflows of the C-MĀIKI project. The design utilizes an "admin dashboard" theme featuring a collapsible primary navigation menu pinned to the left of the screen with content appearing in the main panel. A static header displays session length and the logout button in the upper right corner (Figure 2).

The user interface application is hosted on a linux (Centos7) virtual machine with 4GB of memory and 4 cores. The application is served by the Apache webserver and uses LetsEncrypt Secure Sockets Layer (SSL) certifications to insure traffic is encrypted.

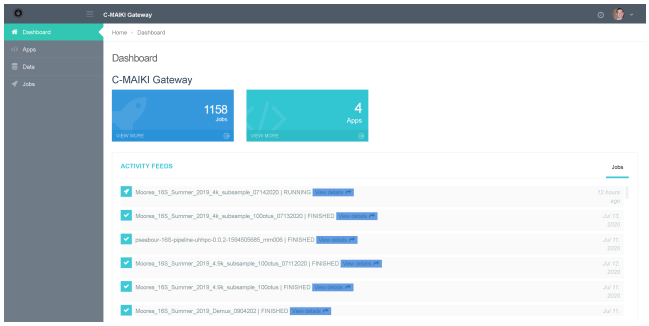

Figure 2: C-MĀIKI gateway dashboard view that provides user the latest activity happening in the gateway and quick access to their recent jobs and statues.

4.3 Data Management and Transfer

The gateway allows users to manage data in a shared storage resource. The data interface appears as a navigable directory structure allowing users to organize (create, delete and move) folders, and navigate the directory structure (Figure 3). Within the folders a user can upload, download, copy, move and rename files and, for supported formats such as images, txt and pdf, even preview file contents. In addition to the C-MĀIKI UI for managing data, the gateway is also integrated with the Globus[1] data transfer service to allow simple, and robust transfers of the large datasets common in microbiome research to and from personal computers and lab servers.

Figure 3: C-MĀIKI gateway file browser that allows users to access and manage their data.

4.4 Sharing And Collaborative Capabilities

For data produced by the C-MĀIKI project, data sharing and dissemination is an important feature. The gateway UI allows researchers and their collaborators to actively share files with their immediate research teams through the "Data" storage interface. This supports download for account users as well as the generation of limited use download links for external collaborators that expire after a set amount of time or number of uses.

4.5 Scalability

Within the MetaFlow|mics pipeline, many of the modules used process each sample independently, making paralleization ideal, especially when analyzing thousands of samples. MetaFlow|mics' use of Netxtflow streamlines parallelization by automatically transferring data between allocated machines, running independent tasks in parallel (or linearly if resources are limited) and collecting results. In the gateway, MetaFlow|mics is configured to leverage the university of Hawaii's Mana High Performance Computing (HPC) cluster currently but could be configured to facilitate deployment on other HPC (SLURM, SGE) and cloud environments (e.g. Google Cloud). In the gateway the selection of required resources, such as queue/partition names can be specified by the user during execution of the pipeline or assigned automatically simplifying the interface for novices. During execution, MetaFlow|mics is configured to automatically parallelize samples by submitting each individual sample processing workflow as a job to the SLURM scheduler, allowing them to run in parallel as resources are available on the system. These processes are then managed by the main MetaFlow|mics process allowing for the C-MAIKI gateway submitted pipelines to run in a hybrid HPC and high throughput manner by submitting to shared and pre-emptable resources to complete job execution in a shorter timeframe. Further, MetaFlow|mics is able to resubmit failed processes and request additional resources from the scheduler. For example, if a process exceeds its wall time or uses too much memory, it will be re-submitted with larger wall time or memory requirements, resuming its computation where it was interrupted.

4.6 Application and Job Management

The C-MĀIKI gateway infrastructure currently supports computations on the University of Hawaii High Performance Compute (HPC) cluster. Tapis is responsible for handling incoming job requests for the user and then staging the data and submitting the MetaFlow|mics pipeline jobs to the HPC scheduler. Each pipeline application is defined using the Nextflow workflow manager [11] that handles all communication with the underlying operating system and ensures reproducibility and traceability of the runs while streamlining resource management and parallelization[2]. The user interface allows researchers to launch the applications and manage the computational jobs from the gateway web interface (Figure 4). The job submission form displays not just fields for parameters and input files but explanations of the parameters and reasonable default values when applicable (Figure 5). This makes MetaFlow|mics pipeline executions simpler for novice users to understand. Users receive notification in the gateway and by email about job status, particularly when they start, complete or fail. Additionally, the traditional workflow of staging, movement, and archiving of software and data to the compute and storage resources is completely managed by Tapis, making application execution fire and forget for the researchers using the gateway.

Job management (Figure 4) provides the status of all current and completed jobs with access to provenance metadata about the job including inputs, outputs, systems, etc. Given these details, a job could be re-run or reproduced by simply re-using its metadata and input files.

5 REPRODUCIBILITY

In a 2016 study [3], 90% of the 1,576 researchers surveyed agreed that reproducibility is an issue in research. Lack of reproducibility is often due to insufficient or unclear method description [5] but

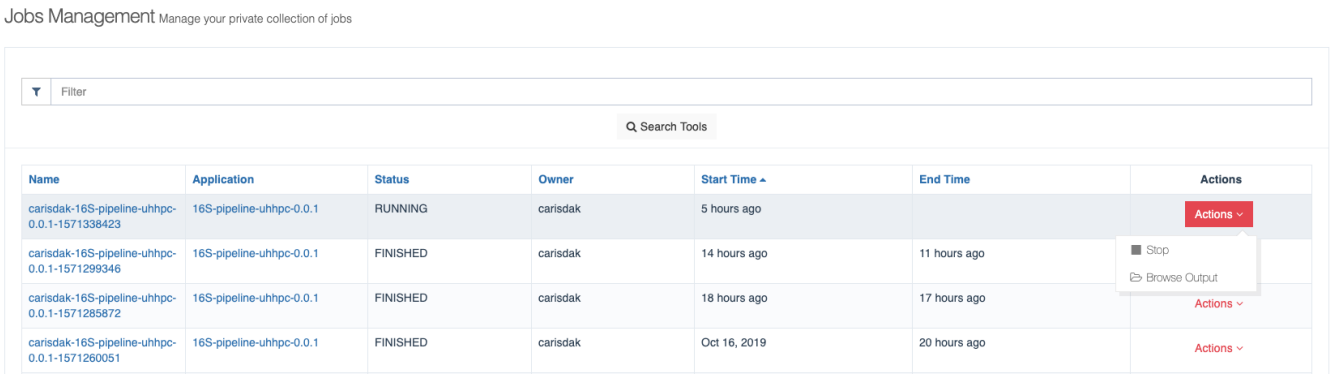

Figure 4: C-MĀIKI gateway job management interface displaying completed and running application executions and actions for viewing job outputs, history and provenance.

can also stem from the version of the Operating System or software in use [6]. Reproducibility is tackled on two fronts by the C-MĀIKI gateway. First, MetaFlow|mics addresses reproducibility in general across environments (OS, architecture, program versions, etc.) in particular by leveraging containerized computing: Details of the computing environment required to run each of the pipelines is described in Dockerfiles that bundle all necessary deployment information, such as operating system type and version and software versions; the docker files are then used to faithfully reproduce any environment as a standalone container. For backward compatibility considerations, previous pipeline versions can be accessed on Dockerhub and the source on GitHub, providing the user with a way to switch to previous run settings if needed. Second, the gateway tracks provenance and provides the metadata on all parameters and inputs for a given computational workflow execution, linking to all the input and output data in the central project repository. The provenance information is stored in the Tapis Job metadata service to ensure reproducibility of the runs and can be accessed via the user interface or the API.

5.1 Provenance

The gateway provides provenance of all application executions by tracking the pipeline parameters, input files, outputs and logs. The gateway stores the metadata about each application execution in the Tapis MongoDB store automatically at the submission of the job and updates through the completion of the execution and archiving of the outputs. In addition, the gateway adds metadata about the application along with the application version and deployment location so in the future it could be re-used to reproduce results as newer software versions are deployed to the gateway this is crucial for ensuring reproducible future runs with the correct software. Lastly, the gateway also tracks the execution start and end times along with the execution resource used, which can be useful if specific hardware configurations or troubleshooting is necessary for an experiment.

6 FUTURE WORK

Future work will include adding additional configurations for XSEDE resources to enable easier deployment of the pipelines to some of these systems. Further, the integration with Jetstream's OpenStack API and commercial cloud offerings for accessing on-demand cloud resources will be pursued. Additionally, the User Inteface is in the process of being updated to ReactJS to leverage the tapis-ui project (https://github.com/tapis-project/tapis-ui) for a more modern and continually supported user interface implementation.

7 CONCLUSION

The key contributions of this paper are the description of the C-MĀIKI gateway's feature and functionality as a purpose-specific science gateway to support microbiome research. The C-MĀIKI gateway provides a modern, easy to use interface that enables the usage of advanced analytical pipelines for processing microbiome data with best practices and state-of-the-art cyberinfrastructure standards to provide simple automated data and analysis management that accelerates science.

8 SOFTWARE AVAILABILITY

The source code for the C-MAIKI gateway is available on Github at https://github.com/UH-CI/cmaiki-gateway and code for the MetaFlow|mics pipeline is available on GitHub at https://github.com/hawaiidatascience/metaflowmics with the documentation compiled and deposited on ReadTheDocs at https://metagenomics-pipelines.readthedocs.io/en/latest/

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation Office of Advanced CyberInfrastructure - Tapis Framrwork #1931439 and #1931575, RII Track 1: 'Ike Wai Securing Hawai'i's Water Future NSF OIA #1557349 and Division of Ocean Sciences #1636402.

REFERENCES

[1] William Allcock, John Bresnahan, Rajkumar Kettimuthu, and Michael Link. 2005. The Globus striped GridFTP framework and server. In *SC'05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*. IEEE, 54–54.

[2] Cedric Arisdakessian, Sean B Cleveland, and Mahdi Belcaid. 2020. MetaFlow|mics: Scalable and Reproducible Nextflow Pipelines for the Analysis of Microbiome Marker Data. In *PEARC20: Proceedings of the Practice and Experience of Advanced Research Computing*. PEARC.

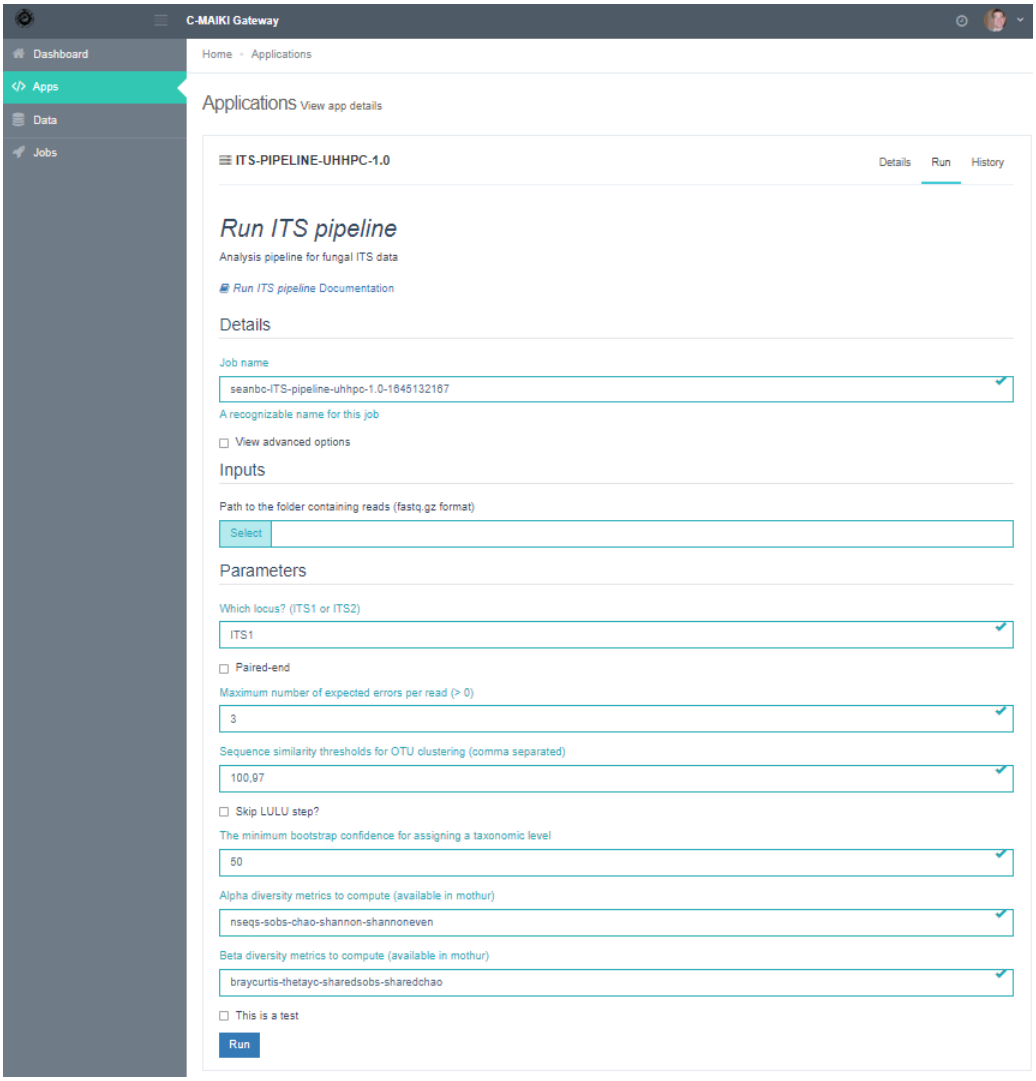

Figure 5: C-MAIKI gateway application submission interface showing the form for the MetaFlow|mics ITS Pipeline for processing fungal sequence data.

[3] Monya Baker. 2016. 1,500 scientists lift the lid on reproducibility. *Nature News* (may 2016).

[4] Michelle Barker, Silvia Delgado Olabarriaga, Nancy Wilkins-Diehr, Sandra Gesing, Daniel S. Katz, Shayan Shahand, Scott Henwood, Tristan Glatard, Keith Jeffery, Brian Corrie, Andrew Treloar, Helen Glaves, Lesley Wyborn, Neil P. Chue Hong, and Alessandro Costa. 2019. The global impact of science gateways, virtual research environments and virtual laboratories. *Future Generation Computer Systems* 95 (2019), 240 – 248. https://doi.org/10.1016/j.future.2018.12.026

[5] Brett K Beaulieu-Jones and Casey S Greene. 2017. Reproducibility of computational workflows is automated using continuous analysis. *Nature Biotechnology* 35, 4 (April 2017), 342–346. https://doi.org/10.1038/nbt.3780

[6] Jayanti Bhandari Neupane, Ram P. Neupane, Yuheng Luo, Wesley Y. Yoshida, Rui Sun, and Philip G. Williams. 2019. Characterization of Leptazolines A–D, Polar Oxazolines from the Cyanobacterium Leptolyngbya sp., Reveals a Glitch with the "Willoughby–Hoye" Scripts for Calculating NMR Chemical Shifts. *Organic Letters* 21, 20 (Oct. 2019), 8449–8453. https://doi.org/10.1021/acs.orglett.9b03216

[7] Evan Bolyen, Jai Ram Rideout, Matthew R Dillon, Nicholas A Bokulich, Christian Abnet, Gabriel A Al-Ghalith, Harriet Alexander, Eric J Alm, Manimozhiyan Arumugam, Francesco Asnicar, and others. 2018. *QIIME 2: Reproducible, interactive, scalable, and extensible microbiome data science.* Technical Report. PeerJ Preprints.

[8] Tilo Buschmann. 2016. DNABarcodes: an R package for the systematic construction of DNA sample tags. *Bioinformatics* 33, 6 (2016), 920–922. https://doi.org/10.1093/bioinformatics/btw759

[9] Leonid V. Bystrykh. 2012. Generalized DNA Barcode Design Based on Hamming Codes. *PLOS ONE* 7, 5 (05 2012), 1–8. https://doi.org/10.1371/journal.pone.0036852

[10] Benjamin J Callahan, Paul J McMurdie, Michael J Rosen, Andrew W Han, Amy Jo A Johnson, and Susan P Holmes. 2016. DADA2: High-resolution sample inference from Illumina amplicon data. *Nature Methods* 13, 7 (July 2016), 581–583. https://doi.org/10.1038/nmeth.3869

[11] Paolo Di Tommaso, Maria Chatzou, Evan W Floden, Pablo Prieto Barja, Emilio Palumbo, and Cedric Notredame. 2017. Nextflow enables reproducible computational workflows. *Nature biotechnology* 35, 4 (2017), 316–319.

[12] Xiaoli Dong, Manuel Kleiner, Christine E Sharp, Erin Thorson, Carmen Li, Dan Liu, and Marc Strous. 2017. Fast and simple analysis of MiSeq amplicon sequencing data with MetaAmp. *Frontiers in microbiology* 8 (2017), 1461.

[13] Rion Dooley and Sean B. Cleveland. 2018. Accelerating Gateway Development with Agave ToGo Webapps and Microsites. 10th International Workshop on Science Gateways (IWSG 2018).

[14] Micah Hamady, Jeffrey J Walker, J Kirk Harris, Nicholas J Gold, and Rob Knight. 2008. Error-correcting barcoded primers for pyrosequencing hundreds of samples

in multiplex. *Nature Methods* 5, 3 (March 2008), 235–237. https://doi.org/10.1038/nmeth.1184

[15] James J. Kozich, Sarah L. Westcott, Nielson T. Baxter, Sarah K. Highlander, and Patrick D. Schloss. 2013. Development of a Dual-Index Sequencing Strategy and Curation Pipeline for Analyzing Amplicon Sequence Data on the MiSeq Illumina Sequencing Platform. *Applied and Environmental Microbiology* 79, 17 (2013), 5112–5120. https://doi.org/10.1128/AEM.01043-13 arXiv:https://aem.asm.org/content/79/17/5112.full.pdf

[16] Katherine A. Lawrence, Michael Zentner, Nancy Wilkins-Diehr, Julie A. Wernert, Marlon Pierce, Suresh Marru, and Scott Michael. 2015. Science gateways today and tomorrow: positive perspectives of nearly 5000 members of the research community. *Concurrency and Computation: Practice and Experience* 27, 16 (2015), 4252–4268. https://doi.org/10.1002/cpe.3526 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpe.3526

[17] Kenneth J Locey and Jay T Lennon. 2016. Scaling laws predict global microbial diversity. *Proceedings of the National Academy of Sciences* 113, 21 (2016), 5970–5975.

[18] Christopher Quince, Anders Lanzen, Russell J. Davenport, and Peter J. Turnbaugh. 2011. Removing Noise From Pyrosequenced Amplicons. *BMC Bioinformatics* 12, 1 (Jan. 2011), 38. https://doi.org/10.1186/1471-2105-12-38

[19] Torbjørn Rognes, Tomáš Flouri, Ben Nichols, Christopher Quince, and Frédéric Mahé. 2016. VSEARCH: a versatile open source tool for metagenomics. *PeerJ* 4 (2016), e2584.

[20] Patrick D Schloss, Sarah L Westcott, Thomas Ryabin, Justine R Hall, Martin Hartmann, Emily B Hollister, Ryan A Lesniewski, Brian B Oakley, Donovan H Parks, Courtney J Robinson, and others. 2009. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl. Environ. Microbiol.* 75, 23 (2009), 7537–7541.

[21] Shengling Wang, Chenyu Wang, Shenling Wang, and Liran Ma. 2018. Big data analysis for evaluating bioinvasion risk. *BMC Bioinformatics* 19, 9 (Aug. 2018), 287. https://doi.org/10.1186/s12859-018-2272-5

[22] N. Wilkins-Diehr, D. Gannon, G. Klimeck, S. Oster, and S. Pamidighantam. 2008. TeraGrid Science Gateways and Their Impact on Science. *Computer* 41, 11 (2008), 32–41.