Application: Transparent, open & sustainable infrastructure for conda-forge and bioconda

Wolf Vollprecht - wolf.vollprecht@quantstack.net
EOSS5: Essential Open Source Software for Science (Cycle 5)

Summary

ID: EOSS5-0000000209
Last submitted: Apr 19 2022 09:11 PM (CEST)

1. Applicant Details

Completed - Apr 19 2022

1. Applicant Details

Complete the following information for the Applicant (required)

The information entered should be for the individual submitting the application who will act as the main person responsible for the application and as its point of contact. **To edit your name or email**, navigate to Account Information by clicking your name in the upper right corner.

Name: Wolf Vollprecht

Email: wolf.vollprecht@quantstack.net

Add your home institution, company, or organization. This does not need to be the organization to which a grant would ultimately be awarded, if selected for funding.

Institution/Affiliation	NumFOCUS / QuantStack

2. Proposal Details

Completed - Apr 19 2022

2. Proposal Details

a. Proposal Title: **Transparent, open & sustainable infrastructure for conda-forge and bioconda**

To edit your proposal title, navigate to the main page; click on the three dots to the right of the application title; and select Rename from the dropdown menu. Proposal title is limited to 60 characters including spaces.

---

b. Amount Requested

Enter requested budget in USD, including indirect costs. This number should be between $100k and $400k over a two year period. Enter whole numbers only (no dollar signs, commas, or cents).

400000

---

c. Proposal Summary/Scope of Work

Provide a short summary of the work being proposed (maximum of 500 words)

The core scientific principle of reproducibility is, in many ways, parallel to the core open-source tenets. In an open-source context, the scientific community can analyse every step of the process, building trust in its effectiveness and contributing to its robustness by identifying bugs when they arise. However, aiming for reproducibility is a complex task involving challenges regarding data provenance and deterministic development environments.

The conda-forge and bioconda projects were founded in 2015 in response to frustrations scientific software users consistently faced when attempting to install system package dependencies. Installing open-source software packages with binary dependencies is frequently a multi-step process involving an intricate sequence of software compilation. The emergence of conda-forge massively reduced the scientific packaging toil by building on transparency, automation, compatibility and open-source principles. As a result, its community has grown exponentially, as has the number of artefacts hosted and downloaded (~18 thousand packages hosted and ~300 million package downloads/month). Such growth has significantly increased the pressure on its underlying infrastructure, tooling, and maintainers' workflow. Besides, this entire infrastructure and community rely on the anaconda.org service, which is not open-source.

To ensure the long-term sustainability of these projects, we propose migrating to fully open-source tooling as follows:

1. Reducing infrastructure technical debt

Conda-forge infrastructure and tooling are distributed across many GitHub repositories, external CI services (Azure DevOps, GitHub Actions, TravisCI, Drone.io, CircleCI), Heroku "dynos" and AWS instances. Many were built as ad-hoc fixes and currently lack documentation or risk mitigation plans. We plan to migrate the configuration and infrastructure provisioning to reproducible, vendor-agnostic tools such as Terraform, complemented with rigorous testing, vulnerability detection, and documentation strategies to enable better security, reliability, and recovery from adverse events.

2. Adopting an OCI-based mirroring strategy

Anaconda.org is the default and sole host for all published and installable scientific packages. Adopting vendor-neutral tooling and standards (such as OCI) will ensure we uphold the core principles of open source and aid the project's long-term sustainability. We also believe that using and building an infrastructure that follows these open principles are the right foundation for more productive and impactful research and education.

3. Development of a maintenance dashboard on Quetz

There is no straightforward way to monitor the operational status of conda-forge's infrastructure. The existing conda-forge.org/status panel is far from giving a comprehensive view of ongoing maintenance tasks, bottlenecks or the overall health of the many bots and infrastructure pieces. Having a detailed picture of the infrastructure and automation tools will significantly improve the maintainers' workflow and aid with identifying critical risks— which is essential to keeping up with the increasing growth and demand from the community. Quetz is chosen as an open-source server for hosting conda packages, thus allowing for increased transparency and extensibility. This would result in the added benefit of centralising the currently scattered-across-repositories packaging metadata in a canonical, API-first, performant-at-scale database, laying the foundation for further infrastructure automation and improvements to the building processes.

d. Value to Biomedical Users

Described the expected value the proposed work to the biomedical research community (maximum of 250 words)

Conda-based packaging has empowered researchers in many fields, allowing them to reduce the time needed to set up and share their working environment. In most cases, conda packages avoid the need to compile from source or ask the IT department to provide a specific library version.

Conda-forge and bioconda are two primary examples of community-driven responses to satisfy the needs of domain-specific packages in the conda ecosystem.
Bioconda alone provides ~9 thousand packages for the life sciences, including bioinformatics, genomics, medical imaging and molecular simulation. It relies on conda-forge to provide its supporting dependencies and provides ~27 thousand packages ready to install across various operating systems and architectures.

Ensuring that both conda-forge and bioconda are sustainable in the long-term is of paramount interest to the whole biomedical community. The proposed work aims to help with conda-forge's most pressing issues threatening its sustainability and ability to meet its vast and diverse community requirements. This work will also allow conda-forge and bioconda to support the principle of a user's right to replicate, i.e. using open source software for infrastructure, adopting vendor-agnostic tooling and APIs and having in-depth technical documentation. Consequently, the projects (and the broader community) will be able to port the infrastructure to any cloud provider (thus avoiding vendor lock-in), replicate the infrastructure and tooling, and adopt a more decentralised approach to scientific packaging and distribution. Therefore, benefiting the open-source, open education and open research ecosystems.

e. Open Source Software Projects

Number of software projects are involved in your proposal (maximum of five):

4

Complete the table with the following information for each software project. If there is no homepage URL, re-enter the main code repository URL.

	Software project name	Main code repository URL	Homepage URL
1	conda-forge	https://github.com/conda-forge	https://conda-forge.org
2	bioconda	https://github.com/bioconda	https://bioconda.github.io/
3	conda-smithy	https://github.com/conda-forge/conda-smithy	https://github.com/conda-forge/conda-smithy
4	quetz	https://github.com/mamba-org/quetz	https://github.com/mamba-org/quetz

f. Landscape Analysis

Briefly describe the other software tools (either proprietary or open source) that the audience for this proposal primarily uses. How do the software project(s) in this proposal compare to these other tools in terms of user base size, usage, and maturity? How do existing tools and the project(s) in this proposal interact? (maximum of 250 words)

The conda ecosystem is the predominant choice in the bioinformatics community, thanks to its extensive selection of projects released via bioconda and conda-forge.
While there are many more package managers available, the alternatives are either platform-specific (e.g. apt, brew, choco), language-specific (e.g. pip for Python, gem for Ruby), or workflow-specific (e.g. Spack for HPC).

Most of the widely available package managers provide little flexibility for users to specify and install specific (and often multiple) versions of a given package. Instead, they adopt patterns such as:

1. The distribution manager defines the versions of the packages on each release cycle (i.e. Linux package managers like apt). This prevents the user from choosing specific package versions and often leaves them with outdated or vulnerable packages.

2. The package manager constantly applies rolling releases or updates to the latest versions, making it extremely challenging for the user to roll back to previous versions (i.e. Arch Linux, Brew).

These approaches result in countless hours worth of human effort lost by researchers and developers who require a particular software configuration to conduct their investigations or verify other research work's reproducibility.

The conda ecosystem is possibly the only one to offer the following guarantees, essential for scientific reproducibility:

1. Choosing which version of a package to install, potentially with different supporting libraries (e.g. Numpy with openblas or mkl)
2. Built-in virtual environments (coexistence of multiple installations)
3. Guaranteed access to previous installations
4. A vast catalogue of available packages across operating systems, architectures and languages

g. Category

Choose the two categories that best describe the software project(s) audience.

	Category
Category 1	Bioinformatics
Category 2	Data management and workflows

h. Previous CZI Funding

Did you previously apply for funding for this or a related proposal under the CZI EOSS program?	Yes
Have you previously received funding for this proposal under the CZI EOSS program?	No