Bioconda: sustainable and comprehensive software distribution for the life sciences

To the Editor: Bioinformatics software comes in a variety of programming languages and requires diverse installation methods. This heterogeneity makes management of a software stack complicated, error-prone, and inordinately time-consuming. Whereas software deployment has traditionally been handled by administrators, ensuring the reproducibility of data analyses[1–3] requires that the researcher be able to maintain full control of the software environment, rapidly modify it without administrative privileges, and reproduce the same software stack on different machines.

The Conda package manager (https://conda.io) has become an increasingly popular means to overcome these challenges for all major operating systems. Conda normalizes software installations across language ecosystems by describing each software with a human readable 'recipe' that defines meta-information and dependencies, as well as a simple 'build script' that performs the steps necessary to build and install the software. Conda builds software packages in an isolated environment, transforming them into relocatable binaries. Importantly, it obviates reliance on system-wide administration privileges by allowing users to generate isolated software environments in which they can manage software versions by project, without generating incompatibilities and side-effects (Supplementary Results). These environments support reproducibility, as they can be rapidly exchanged via files that describe their installation state. Conda is tightly integrated into popular solutions for reproducible data analysis such as Galaxy[4], bcbio-nextgen (https://github.com/chapmanb/bcbio-nextgen), and Snakemake[5]. To further enhance reproducibility guarantees, Conda can be combined with container or virtual machine-based approaches and archive facilities such as Zenodo (Supplementary Results). Finally, although Conda provides many commonly used packages by default, it also allows users to optionally include additional, community-managed repositories of packages (termed channels).

To unlock the benefits of Conda for the life sciences, we present the Bioconda project (https://bioconda.github.io). The Bioconda project provides over 3,000 Conda software packages for Linux and macOS. Rapid turnaround times (Supplementary Results) and extensive documentation (https://bioconda.github.io/contributing.html) have led to a growing community of over 200 international scientists working in the project (Supplementary Results). The project is led by a core team, which is complemented by interest groups for particular language ecosystems. Unlimited (in time and space) storage for generated packages is donated by Anaconda Inc. All other used infrastructure is free of charge. Bioconda provides packages from various language ecosystems such as Python, R (CRAN and Bioconductor), Perl, Haskell, Java, and C/C++ (Fig. 1a). Many of the packages have complex dependency structures that require various manual steps for installation when not relying on a package manager like Conda (Supplementary Results). With over 6.3 million downloads, Bioconda has become a backbone of bioinformatics infrastructure that is used heavily across all language ecosystems (Fig. 1b). It is complemented by the conda-forge project (https://conda-forge.github.io), which hosts software not specifically related to the biological sciences. This separation has proven beneficial, because the focused nature of the Bioconda community allows for fast turnaround times and support when a user needs to contribute

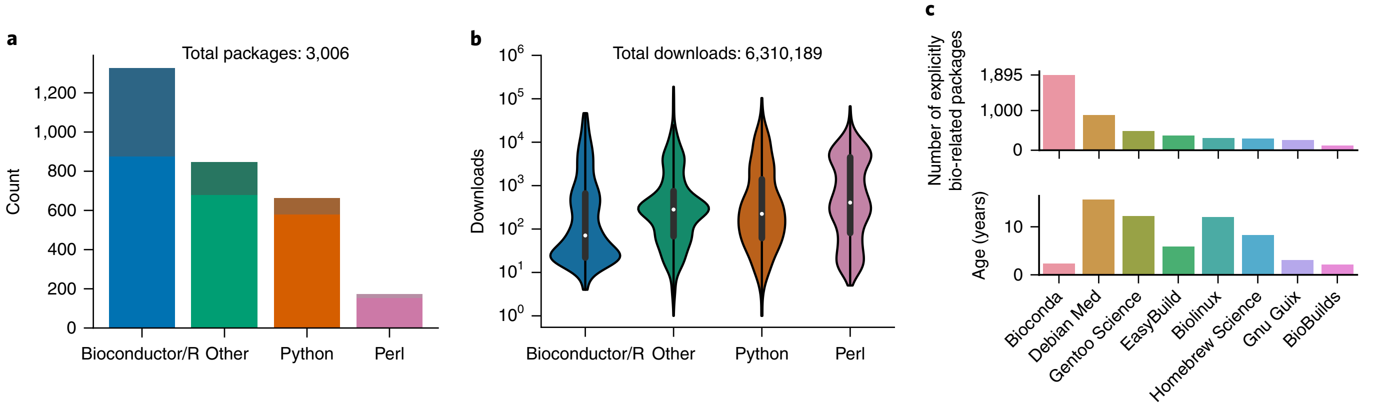

Fig. 1 | Package numbers and usage. a, Package count per language ecosystem (saturated colors on the lower portions of the bars represent explicitly life-science-related packages). **b**, Distribution of per-package downloads, separated by language ecosystem. The term "other" encompasses all packages that do not fall into one of the specific categories named. White dots represent the mean; dark bars represent the interval between upper and lower quartiles. **c**, Comparison of the number of explicitly life-science-related packages in Bioconda with that in Debian Med (https://www.debian.org/devel/debian-med), Gentoo Science Overlay (category sci-biology; https://github.com/gentoo/sci), EasyBuild (module bio; https://easybuilders.github.io/easybuild), Biolinux[6], Homebrew Science (tag bioinformatics; https://brew.sh), GNU Guix (category bioinformatics; https://www.gnu.org/s/guix), and BioBuilds (https://biobuilds.org). The lower graph shows the project age since the first release or commit. Statistics obtained 25 October 2017.

packages or fix problems. Nevertheless, the two projects collaborate closely, and the Bioconda team maintains over 500 packages hosted by conda-forge.

Bioconda is not the only effort to distribute bioinformatics software (Fig. 1c). The alternatives can be categorized into system-wide (Debian-Med, Genotoo Science, Biolinux, and Homebrew) and per-user (EasyBuild, GNU Guix, and BioBuilds) installation mechanisms. The system-wide approaches lack the ability to put the scientist in control of the installed software stack, and thus do not meet the requirements for reproducibility outlined above. All per-user-based approaches provide a similar feature set (BioBuilds is also using the Conda package manager). However, among all available approaches, Bioconda, despite being the most recent, is by far the most comprehensive, with thousands of software libraries and tools that are maintained by hundreds of international contributors (Fig. 1c).

For reproducible data science, it is crucial that software libraries and tools be provided via an easy-to-use, unified interface, so that they can be easily deployed and sustainably managed. With its ability to maintain isolated software environments, integration into major workflow management systems, and lack of requirement for any administration privileges for use, the Conda package manager is the ideal tool to ensure sustainable and reproducible software management. Bioconda packages have been well received by the community, with over six million downloads so far. We invite everybody to join the Bioconda community, participate in maintaining or publishing new software, and work toward the goal of a central, comprehensive, and language-agnostic collection of easily installable software for the life sciences.

Reporting Summary. Further information on experimental design is available in the Nature Research Reporting Summary linked to this article.

Data availability. Data and code underlying the presented results are enclosed in a Snakemake workflow archive available at https://doi.org/10.5281/zenodo.1068297. The archive can also be used to automatically reproduce all results and figures presented in this paper. ❐

Björn Grüning[1,12], Ryan Dale[2,12], Andreas Sjödin[3,4], Brad A. Chapman[5], Jillian Rowe[6], Christopher H. Tomkins-Tinch[7,8], Renan Valieris[9], Johannes Köster[10,11]* and The Bioconda Team[13]

[1]Bioinformatics Group, Department of Computer Science, University of Freiburg, Freiburg, Germany. [2]Laboratory of Cellular and Developmental Biology, National Institute of Diabetes and Digestive and Kidney Diseases, US National Institutes of Health, Bethesda, MD, USA. [3]Division of CBRN Security and Defence, FOI–Swedish Defence Research Agency, Umeå, Sweden. [4]Department of Chemistry, Computational Life Science Cluster (CLiC), Umeå University, Umeå, Sweden. [5]Harvard T.H. Chan School of Public Health, Boston, MA, USA. [6]Center for Genomics and Systems Biology, Genomics Core,, NYU Abu Dhabi,, Abu Dhabi,, United Arab Emirates. [7]Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA, USA. [8]Broad Institute of MIT and Harvard, Cambridge, MA, USA. [9]Laboratory of Bioinformatics and Computational Biology, A. C. Camargo Cancer Center, São Paulo, Brazil. [10]Algorithms for Reproducible Bioinformatics, Genome Informatics, Institute of Human Genetics, University Hospital Essen, University of Duisburg–Essen, Essen, Germany. [11]Medical Oncology, Dana Farber Cancer Institute, Harvard Medical School, Boston, MA, USA. [12]These authors contributed equally: Björn Grüning and Ryan Dale. [13]A full list of authors and affiliations is available as Supplementary Table 1.
**e-mail: johannes.koester@uni-due.de*

Published online: 2 July 2018
https://doi.org/10.1038/s41592-018-0046-7

References

1. Mesirov, J. P. *Science* **327**, 415–416 (2010).
2. Baker, M. *Nature* **533**, 452–454 (2016).
3. Munafò, M. R. et al. *Nat. Hum. Behav.* **1**, 0021 (2017).
4. Afgan, E. et al. *Nucleic Acids Res.* **44**, W3–W10 (2016).
5. Köster, J. & Rahmann, S. *Bioinformatics* **28**, 2520–2522 (2012).
6. Field, D. et al. *Nat. Biotechnol.* **24**, 801–803 (2006).

Acknowledgements

We thank all contributors, the conda-forge team, and Anaconda Inc. for excellent cooperation. Further, we thank Travis CI (https://travis-ci.com) and Circle CI (https://circleci.com) for providing free Linux and macOS computing capacity. Finally, we thank ELIXIR (https://www.elixir-europe.org) for constant support and donation of staff. This work was supported by the Intramural Program of the National Institute of Diabetes and Digestive and Kidney Diseases, US National Institutes of Health (R.D.), the Netherlands Organisation for Scientific Research (NWO) (VENI grant 016. Veni.173.076 to J.K.), the German Research Foundation (SFB 876 to J.K.), and the NYU Abu Dhabi Research Institute for the NYU Abu Dhabi Center for Genomics and Systems Biology, program number CGSB1 (grant to J.R. and A. Yousif).

Author contributions

J.K. and R.D. wrote the manuscript and conducted the data analysis. K. Beauchamp, C. Brueffer, B.A.C., F. Eggenhofer, B.G., E. Pruesse, M. Raden, J.R., D. Ryan, I. Shlyakter, A.S., C.H.T.-T., and R.V. (in alphabetical order) contributed to writing of the manuscript. D.A. Søndergaard supervised student programmers on writing Conda package recipes and maintaining the connection with ELIXIR. All other members of the Bioconda Team contributed or maintained recipes (author order was determined by the number of commits in October 2017).

Competing interests

The authors declare no competing interests.

Additional Information

Supplementary information is available for this paper at https://doi.org/10.1038/s41592-018-0046-7.

nature research

Corresponding author(s): Johannes Köster

□ Initial submission □ Revised version ☒ Final submission

Life Sciences Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form is intended for publication with all accepted life science papers and provides structure for consistency and transparency in reporting. Every life science submission will use this form; some list items might not apply to an individual manuscript, but all fields must be completed for clarity.

For further information on the points included in this form, see Reporting Life Sciences Research. For further information on Nature Research policies, including our data availability policy, see Authors & Referees and the Editorial Policy Checklist.

▸ Experimental design

1. Sample size

Describe how sample size was determined.

No sampling was performed. All statistics have been calculated on the entire set of packages.

2. Data exclusions

Describe any data exclusions.

No data was excluded. Statistics were obtained at Oct. 25, 2017.

3. Replication

Describe whether the experimental findings were reliably reproduced.

Since no experiments were performed, replication is not applicable.

4. Randomization

Describe how samples/organisms/participants were allocated into experimental groups.

The presented work does not involve any experiments. All provided statistics are just a description of the Bioconda resource.

5. Blinding

Describe whether the investigators were blinded to group allocation during data collection and/or analysis.

The presented work does not involve any experiments. Data acquisition was done in an unbiased and automatic way by parsing package statistics.

Note: all studies involving animals and/or human research participants must disclose whether blinding and randomization were used.

6. Statistical parameters

For all figures and tables that use statistical methods, confirm that the following items are present in relevant figure legends (or in the Methods section if additional space is needed).

n/a	Confirmed	
☒	□	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement (animals, litters, cultures, etc.)
☒	□	A description of how samples were collected, noting whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	□	A statement indicating how many times each experiment was replicated
☒	□	The statistical test(s) used and whether they are one- or two-sided (note: only common tests should be described solely by name; more complex techniques should be described in the Methods section)
☒	□	A description of any assumptions or corrections, such as an adjustment for multiple comparisons
☒	□	The test results (e.g. *P* values) given as exact values whenever possible and with confidence intervals noted
☒	□	A clear description of statistics including central tendency (e.g. median, mean) and variation (e.g. standard deviation, interquartile range)
☒	□	Clearly defined error bars

See the web collection on statistics for biologists for further resources and guidance.

Software

Policy information about availability of computer code

7. Software

Describe the software used to analyze the data in this study.

Analyses were conducted using a fully reproducible Snakemake 4.8.0 workflow available under https://doi.org/10.5281/zenodo.1068298.

For manuscripts utilizing custom algorithms or software that are central to the paper but not yet described in the published literature, software must be made available to editors and reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). *Nature Methods* guidance for providing algorithms and software for publication provides further information on this topic.

Materials and reagents

Policy information about availability of materials

8. Materials availability

Indicate whether there are restrictions on availability of unique materials or if these materials are only available for distribution by a for-profit company.

no unique materials were used

9. Antibodies

Describe the antibodies used and how they were validated for use in the system under study (i.e. assay and species).

no antibodies were used

10. Eukaryotic cell lines

a. State the source of each eukaryotic cell line used.

no eukaryotic cell lines were used

b. Describe the method of cell line authentication used.

no eukaryotic cell lines were used

c. Report whether the cell lines were tested for mycoplasma contamination.

no eukaryotic cell lines were used

d. If any of the cell lines used are listed in the database of commonly misidentified cell lines maintained by ICLAC, provide a scientific rationale for their use.

no commonly misidentified cell lines were used

Animals and human research participants

Policy information about studies involving animals; when reporting animal research, follow the ARRIVE guidelines

11. Description of research animals

Provide details on animals and/or animal-derived materials used in the study.

no animals were used

Policy information about studies involving human research participants

12. Description of human research participants

Describe the covariate-relevant population characteristics of the human research participants.

the study did not involve human research participants