# Packaging and containerizing of bioinformatics software: advances, challenges, and opportunities

*This manuscript ([permalink]) was automatically generated from [Mangul-Lab-USC/pkg-manager-review@e650028] on August 24, 2020.*

## Unsorted list of authors


- **Sharon Waymost**
  0000-0003-1176-5386 · sbpw
  Computer Science Department, Samueli School of Engineering, University of California, Los Angeles

- **Serghei Mangul**
  0000-0003-4770-3443 · smangul1
  Department of Clinical Pharmacy, School of Pharmacy, University of Southern California

- **Neha Rajkumar**

  Department of Bioengineering, Samueli School of Engineering, University of California, Los Angeles

- **Ram Ayyala**

  Department of Neuroscience, School of Life Sciences, University of California Los Angeles

- **Heng Li**

  None

- **Nathan LaPierre**

  None

- **André Santos**

  None

- **Titus Brown**

  University of California, Davis

- **Casey S. Greene**
  0000-0001-8713-9213
  Department of Systems Pharmacology and Translational Therapeutics, Perelman School of Medicine, University of Pennsylvania; Childhood Cancer Data Lab

- **Brendan Lawlor**
  0000-0002-2250-0669 · blawlor · blawlor
  Department of Computer Science, Cork Institute of Technology, Cork, Ireland.

- **Qiyang Hu**

  None

- **Jaqueline J. Brito**
  0000-0002-7158-3253 · jaquejbrito · jaquejbrito
  Department of Clinical Pharmacy, School of Pharmacy, University of Southern California

- **Thiago Mosqueiro**

  None

## Abstract

## Introduction

The rapid advancement of omics and sequencing technologies has led to an accelerated growth of genomic data. Driven by this data, there are similar increases in the number of available bioinformatics software tools, often containing novel and diverse algorithms of particular interest to biomedical researchers.[1,2]. Biomedical researchers wanting to use this software often have access to high-performance clusters ("clusters"), but lack the operating system-level permissions and advanced skills required to install and run the software. None of the necessary skills are currently included in the traditional life science curriculum at major universities. Usability is further hindered by the absence of any standardization and the wide variety of software development tools employed. This is directly observable in many academic software tools lacking a user-friendly interface[5].

As the dependence of biomedical researchers on computational software continues to increase, software developers need to consider more user-friendly distribution and install methods. Modern software distribution, having already faced user-friendliness issues, is already becoming more reliant on platforms such as package managers[6,7] and containers[8]. Both promise to simplify software development while increasing the usability and reproducibility of biomedical research[8,9].

Package managers first appeared nearly thirty years ago as software developers sought to streamline the entire installation process. To install via package manager, the user must only specify the desired software, called a "package;" the download, installation, configuration, and dependencies are all handled by the package manager. Many operating systems have built-in package managers (e.g. APT), which may not be available to cluster users, while others must be downloaded and installed by the user. Some package managers (e.g. Conda[6]) are programming language agnostic, while others are designed for a particular language (e.g pip[10]).

A more recent software distribution solution is containerization. The end-user downloads a container "image" that includes the software, dependencies, and anything else necessary to run the software. Though the imaged software is not typically itself installed, many do require the installation of containerization runtime software. When run, the runtime software creates a consistent, isolated sandbox environment, then runs the imaged software inside the sandbox[11]. This sandbox design makes the images both highly portable (compatible with different computers) and easily shareable (transferable between different computers), which has already led to wide adaptation in bioinformatics[7,12].

In addition to the ease of installation and use offered by both package managers and containers, such platforms must also meet the biomedical community's need for compatibility with high performance clusters. However, the relative performance of these package managers and containers remains unknown. Our review summarizes developing practices across the most common package managers and containers, while discussing the challenges, advantages, and limitations of using them from both the developer and the user perspectives. By taking a survey of all the available package managers and container software, there is now a comprehensive list of the different attributes of each one, informing the community so that people can make educated decisions about which package manager or container makes the most sense for their project. We also propose principles that can make packaging and containerizing of bioinformatics software more sustainable and reproducible, ultimately increasing the usability of bioinformatics software.

## Discussion

### Existing problems with software distribution and installation

The installation process of bioinformatics research software is typically a multi-step process, starting with the end-user locating and downloading the software. Next is the actual installation, during which the end-user must determine what dependencies are missing and resolve them by installing the required software[5]. Even in cases where the end-user is familiar with this process, they are typically installing on high-performance clusters where they are constrained by user-level permissions that prevent them from following standard installation procedures.

- root access limitations
- reproducibility of findings
- version conflicts
- dependency resolution

### Definitions and explanations of distribution system types

- package managers
    - definition

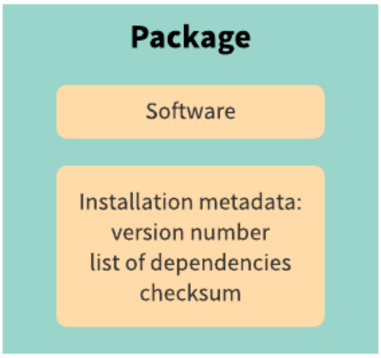
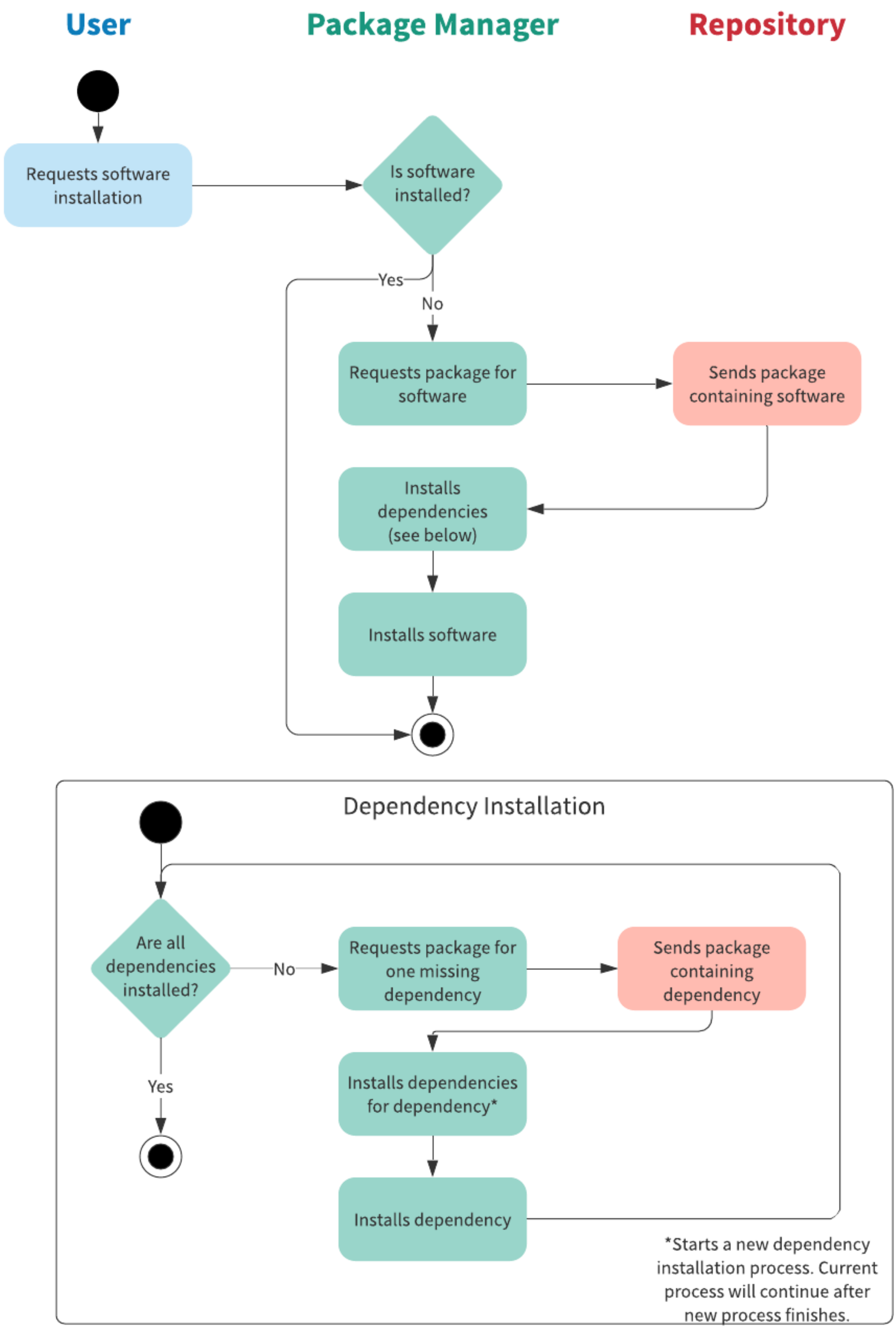



**Figure 1: Schematic showing standard package contents.**




- benefits for the developer - mature technology - higher degree of familiarity - allows dependency specification (including versions) -limitations for the developer - can't always use to install missing dependencies for end-user - benefits for the end-user - package size is minimal (dependencies aren't duplicated) - installs missing dependencies - limitations for the end-user - not always accessible (unless admin user) - can't install multiple versions of same software - containerization - definition

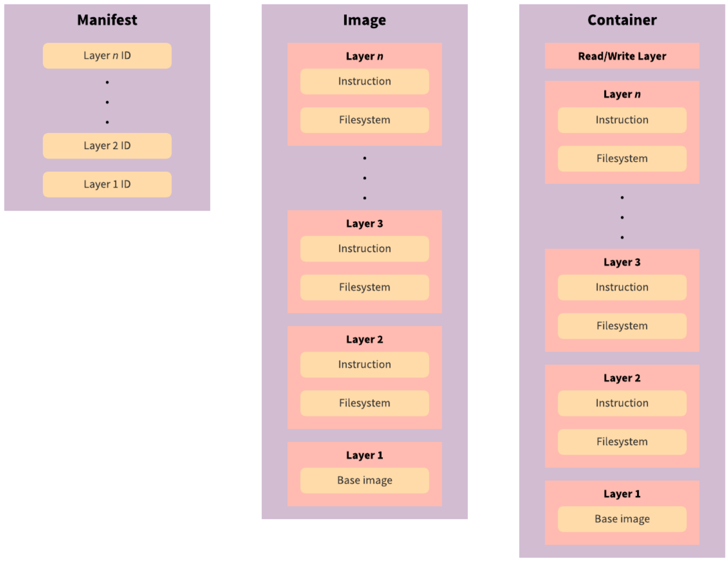
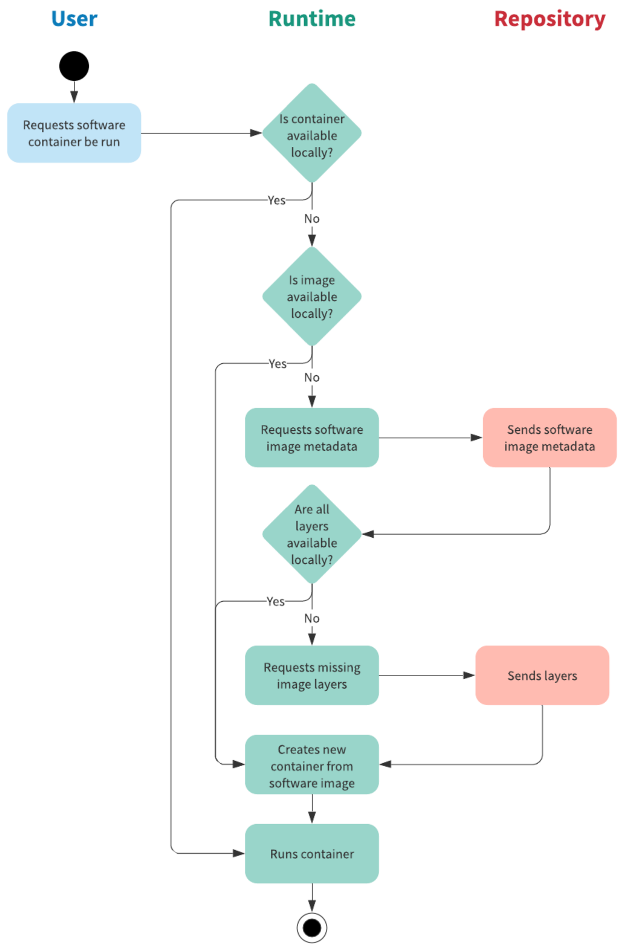



**Figure 2: Schematic showing standard containerization objects.**




- benefits for the developer - include specific versions of dependencies - known running environment - fewer test variables - reproducibility of results - limitations for the developer - learn a new system instead of focusing on research - benefits for the end-user - no installation (except possible runtime) - no dependency issues - sandbox provides computer system security - limitations for the end-user - container size - duplication of dependencies - root access requirement to install runtime - configuration in cluster - centralized repositories - definition - benefits - known download site - hosting is taken of - limitations - repo specific restrictions

## Glossary

## Acknowledgements

## Author Contributions

## References

# Tables

| Distribution System Name | URL | Publication | Type | License |
|---|---|---|---|---|
| AppImage | https://appimage.org | - | containerization | MIT |
| APT | https://wiki.debian.org/Apt | - | package manager | GNU GPL 2+ |
| Bioconda | https://bioconda.github.io | Grüning et al, 2018 | package manager | MIT |
| Bioconductor | https://www.bioconductor.org | Gentleman et al, 2004 | package manager | MIT |
| conda | https://docs.conda.io/en/latest/ | - | package manager | 3-Clause BSD |
| CRAN | https://cran.r-project.org/index.html | - | package manager | GNU GPL |
| Docker | https://www.docker.com | - | containerization | Apache 2.0 |
| Easybuild | https://easybuilders.github.io/easybuild/ | Hoste et al, 2012 | package manager | GNU GPL 2 |
| Flatpak | https://flatpak.org | - | containerization | LGPL |
| GNU Guix | https://guix.gnu.org | Courtès, 2013 | package manager | GNU AGPL |
| Homebrew | https://brew.sh | - | package manager | 2-Clause BSD |
| pip | https://pypi.org/project/pip/ | - | package manager | MIT |
| Singularity | https://sylabs.io | - | containerization | 3-Clause BSD |
| Snap | https://snapcraft.io | - | containerization | propriertary |
| Spack | https://spack.io | Gamblin et al, 2015 | package manager | MIT or Apache |
| Vagrant | https://www.vagrantup.com | - | virtual machine | MIT |
| yum | http://yum.baseurl.org | - | package manager | |
| Zero Install | https://0install.net | - | package manager | GNU LGPL 2.1+ |

| Distribution System Name | Supported Operating Systems | Supported Languages | Root to Install | Root to Run |
|---|---|---|---|---|
| AppImage | Linux | any | n/a | no |
| APT | Debian, Ubuntu | any | yes | yes |
| Bioconda | Linux, macOS, Windows | any | no | no |
| Bioconductor | Linux, macOS, Windows | R | no | no |
| conda | Linux, macOS, Windows | any | no | no |
| CRAN | Linux, macOS, Windows | R | no | no |
| Docker | Linux, macOS, Windows | any | yes | no |
| Easybuild | Linux | any | no | no |
| Flatpak | Linux | any | no | no |
| GNU Guix | Linux | any | no | no |
| Homebrew | Linux, macOS | any | no | no |
| pip | Linux, macOS, Windows | Python | no | no |
| Singularity | Linux, macOS | any | yes | no |
| Snap | Linux | any | yes | no |
| Spack | Linux, macOS | any | no | no |
| Vagrant | Linux, macOS, Windows | any | yes | |
| yum | Linux, macOS, Windows | any | no | yes |
| Zero Install | Linux, macOS, Windows | any | no | no |

| Distribution System Name | First Release | Latest Release | Age in Years | Number of Releases | Number of Tools | Number of Bio Tools |
|---|---|---|---|---|---|---|
| AppImage | 2014-01-24 | 2020-06-01 | 7 | 121 | | |
| APT | 1998-03-31 | 2020-05-08 | 22 | 362 | | |
| Bioconda | 2014-01-24 | 2016-09-06 | 7 | 39 | | |
| Bioconductor | 2002-05-01 | 2020-04-28 | 17 | 37 | | |
| conda | 2014-01-24 | 2020-04-13 | 6 | 261 | | |
| CRAN | 1997-04-23 | 2020-02-29 | 22 | 29 | | |
| Docker* | 2013-03-23 | 2020-06-01 | 7 | 121 | | |
| Easybuild | 2012-11-09 | 2020-04-14 | 7 | 51 | | |
| Flatpak | 2015-03-23 | 2020-04-03 | 5 | 128 | | |
| GNU Guix | 2012-07-07 | 2020-04-15 | 7 | 23 | | |
| Homebrew | 2009-05-20 | 2020-05-04 | 10 | 155 | | |
| pip | 2009-01-20 | 2020-04-28 | 11 | 81 | | |

| Distribution System Name | First Release | Latest Release | Age in Years | Number of Releases | Number of Tools | Number of Bio Tools |
|---|---|---|---|---|---|---|
| Singularity | 2012-07-07 | 2020-04-15 | 7 | 23 | | |
| snapd | 2014-12-09 | 2020-07-15 | 5 | 232 | | |
| Spack | 2014-07-09 | 2020-04-15 | 5 | 27 | | |
| Vagrant | | | | | | |
| yum** | 2002-06-08 | 2011-06-28 | 18 | 221 | | |
| Zero Install | 2005-02-04 | 2020-05-04 | 15 | 145 | | |

*Docker Engine

**need to find someone with a redhat license who can confirm numbers

| Distribution System Name | Official Repository Name | Repository URL |
|---|---|---|
| AppImage | AppImageHub | https://appimage.github.io/apps/ |
| APT | - | - |
| Bioconda | bioconda channel | https://github.com/bioconda/bioconda-recipes |
| Bioconductor | - | https://www.bioconductor.org/packages/release/BiocViews.html#___Software |
| conda | - | https://repo.anaconda.com/pkgs/ |
| CRAN | - | https://cran.r-project.org/web/packages/available_packages_by_name.html |
| Docker | Docker Hub | https://hub.docker.com |
| Easybuild | | |
| Flatpak | Flathub | https://flathub.org |
| GNU Guix | - | https://guix.gnu.org/packages/ |
| Homebrew | Homebrew Formulae | https://formulae.brew.sh |
| pip | Python Package Index (PyPI) | https://pypi.org |
| Singularity | Singularity Hub | https://singularity-hub.org |
| Snap | Snapcraft | https://snapcraft.io/store |
| Spack | - | - |
| Vagrant | Vagrant Cloud | https://app.vagrantup.com/boxes/search |
| yum | - | - |
| Zero Install | - | https://apps.0install.net |