



HAL
open science

BioShaDock: a community driven bioinformatics shared Docker-based tools registry

François Moreews, Olivier Sallou, Hervé Ménager, Yvan Le Bras, Cyril
Monjeaud, Christophe Blanchet, Olivier Collin

► **To cite this version:**

François Moreews, Olivier Sallou, Hervé Ménager, Yvan Le Bras, Cyril Monjeaud, et al.. BioShaDock: a community driven bioinformatics shared Docker-based tools registry. F1000Research, 2015, 10.12688/f1000research.7536.1 . hal-01243520

HAL Id: hal-01243520

<https://inria.hal.science/hal-01243520v1>

Submitted on 15 Dec 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



SOFTWARE TOOL ARTICLE

BioShaDock: a community driven bioinformatics shared Docker-based tools registry [version 1; referees: awaiting peer review]

François Moreews¹, Olivier Sallou², Hervé Ménager³, Yvan Le bras², Cyril Monjeaud², Christophe Blanchet², Olivier Collin⁴

¹Genscale team, IRISA, Rennes, France

²Genouest Bioinformatics Facility, University of Rennes 1/IRISA, Rennes, France

³Centre d'Informatique pour la Biologie, C3BI, Institut Pasteur, Paris, France

⁴French Institute of Bioinformatics, CNRS IFB-Core, Gif-sur-Yvette, France

v1 First published: 14 Dec 2015, 4:1443 (doi: [10.12688/f1000research.7536.1](https://doi.org/10.12688/f1000research.7536.1))
Latest published: 14 Dec 2015, 4:1443 (doi: [10.12688/f1000research.7536.1](https://doi.org/10.12688/f1000research.7536.1))

Abstract

Linux container technologies, as represented by Docker, provide an alternative to complex and time-consuming installation processes needed for scientific software. The ease of deployment and the process isolation they enable, as well as the reproducibility they permit across environments and versions, are among the qualities that make them interesting candidates for the construction of bioinformatic infrastructures, at any scale from single workstations to high throughput computing architectures. The Docker Hub is a public registry which can be used to distribute bioinformatic software as Docker images. However, its lack of curation and its genericity make it difficult for a bioinformatics user to find the most appropriate images needed. BioShaDock is a bioinformatics-focused Docker registry, which provides a local and fully controlled environment to build and publish bioinformatic software as portable Docker images. It provides a number of improvements over the base Docker registry on authentication and permissions management, that enable its integration in existing bioinformatic infrastructures such as computing platforms. The metadata associated with the registered images are domain-centric, including for instance concepts defined in the EDAM ontology, a shared and structured vocabulary of commonly used terms in bioinformatics. The registry also includes user defined tags to facilitate its discovery, as well as a link to the tool description in the ELIXIR registry if it already exists. If it does not, the BioShaDock registry will synchronize with the registry to create a new description in the Elixir registry, based on the BioShaDock entry metadata. This link will help users get more information on the tool such as its EDAM operations, input and output types. This allows integration with the ELIXIR Tools and Data Services Registry, thus providing the appropriate visibility of such images to the bioinformatics community.

Open Peer Review

Referee Status: *AWAITING PEER*

REVIEW

Discuss this article

Comments (0)



This article is included in the **Container Virtualization in Bioinformatics** channel.

Corresponding author: François Moreews (fmoreews@irisa.fr)

How to cite this article: Moreews F, Sallou O, Ménager H *et al.* **BioShaDock: a community driven bioinformatics shared Docker-based tools registry [version 1; referees: awaiting peer review]** *F1000Research* 2015, **4**:1443 (doi: [10.12688/f1000research.7536.1](https://doi.org/10.12688/f1000research.7536.1))

Copyright: © 2015 Moreews F *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution Licence](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Grant information: Funding was provided from the Western France e-science project supported by Brittany and Pays de la Loire regions (e-Biogenouest/052012).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: No competing interests were disclosed.

First published: 14 Dec 2015, **4**:1443 (doi: [10.12688/f1000research.7536.1](https://doi.org/10.12688/f1000research.7536.1))

Introduction

The life sciences are becoming more and more digital and nowadays data analysis methods represent a key factor of the discovery process. In the case of bioinformatics, software is widely provided by the research community. Developers favor open source approaches and many software tools are available online. It is commonly agreed that such a distributed and free creation process accelerates discoveries in the life sciences^{1,2}. However, this view must be nuanced, as multiple factors still hinder the discovery, integration, and maintenance of these software tools.

First, domains such as genomics, where technological innovation leads to an exponential growth of data to analyse, also generate an ever-increasing number of new software methods. However, the discovery of new interesting tools by potential users remains limited by unstructured descriptions, lack of metadata and deprecated source codes. In this context, dedicated search engines like the ELIXIR Tools and Data Services Registry^{3,4} (hereafter referred to as the “ELIXIR registry”) have emerged as a potential solution to search, find and locate available and maintained tools.

Secondly, the implementation methods of bioinformatic software are heterogeneous and their deployment requires multiple technical skills. The **installation process** is therefore expensive, in terms of human resources. It is worth recalling that the cost in supporting operating systems and hardware diversity can be high, the code compilation process is error prone and the required software dependencies are often conflicting with installed libraries. Consequently, the audience of a software can be limited to highly motivated and technical users or large bioinformatics facilities. The recent development of user-friendly data analysis environments like Galaxy⁵ ease access for biologists and bio-analysts to bioinformatic tools. These software workbenches provide a generic web user interface for command line based scientific applications, but do not solve the tools’ deployment issue. Even if the task can be submitted inside a container, it is the tool designer’s responsibility to provide a readily deployable component⁶ and the proportion of container based components in repositories such as the Galaxy Toolsheds⁷ is currently low.

Finally, traditional academic publishing and funding processes emphasize the production of software with short-term goals, these being the publication of the method and/or results. Such an environment does not favor a software engineering-oriented approach to software development⁸, and this affects directly the portability and maintainability of the software products⁹. This in turn impacts the reproducibility of analyses, experiments or benchmarks described in published articles. However, even if various emerging initiatives are developing frameworks^{10–12} to enable a new kind of “executable format” of scientific publication, few journals have an innovative publishing policy that includes the long term storage of the source codes on a dedicated public web platform.

Nevertheless, today containerization brings new pragmatic solutions. Linux containers are a mature technology that has the potential to dramatically facilitate scientific software deployment and analysis reproducibility. Docker, one of the most popular container solutions^{13,14}, is now used in a variety of computation environments,

from commercial clouds¹⁵ to clusters with dedicated middleware¹⁶. It has been positively evaluated for data intensive computation, a recent study showing that the performance of bioinformatic workflows composed by medium or long running tasks are only very slightly affected by containerization¹⁷.

Container technology has the potential to impact audiences, developers and end-users. In the scientific field, it can effectively improve reproducibility, ease deployment and facilitate the building of software collections and search engines dedicated to a specific scientific domain or topic.

For these reasons, we created the BioShaDock registry that promotes the use of container technologies in bioinformatics. The BioShaDock registry provides a web entry point to deploy, search and discover ready to use bioinformatics tools, encapsulated in Docker containers.

Future works will focus on better integration with domain-centric registries as well as bioinformatic integrated environments, to enable the seamless discovery, integration, and execution of the BioShaDock containers. Our project will also greatly benefit from discussions with other existing bioinformatic container initiatives.

Methods

Registration

BioShaDock is a web server based system that allows the description, registration and automated building of Docker images (Figure 1). These images are publicly available on the web server for search, download and execution. Users can authenticate using local LDAP or Google/GitHub credentials. LDAP users have the possibility to push new images. External users (Google, etc.) can request those privileges by contacting the support team. This mechanism allows non local users to have access to the registry to provide new tools while keeping a controlled access on the submission of new tools to the registry, where contributions are based on trust.

Once authenticated, the user can proceed to the **registration** of a Docker container. The information required includes:

- the **set of instructions to build the image**, i.e. the Dockerfile and the associated source code. These can be provided by pasting directly the Dockerfile contents in the web interface, by pointing to a Git repository that contains the Dockerfile and the source code, or by pointing to the source code repository and manually providing the Dockerfile. In the case of Git repository registration, it is also possible to configure the branch and location of the Dockerfile in the repository.
- **additional metadata** which is required to describe the contents of the image in scientific terms to its potential users. Such metadata includes for instance free tags, as well as EDAM¹⁸ terms.

Following the completion of container registration, the image construction and integration steps (Figure 2) are automatically run on a dedicated server. The trigger of a new build is based on Dockerfile

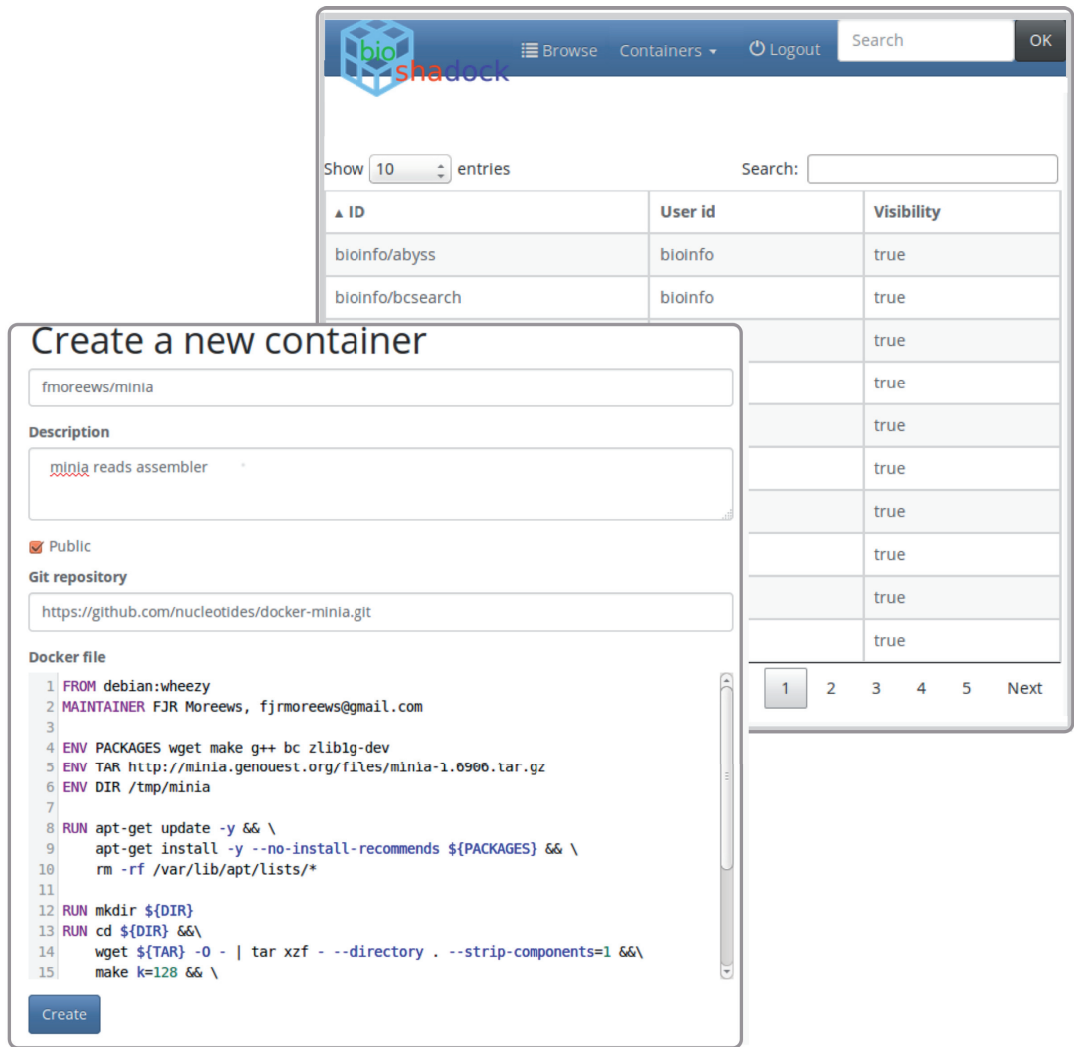


Figure 1. The BioShaDock web interface. The interface enables the creation of Dockerfiles and allows to search the repository using full text queries.

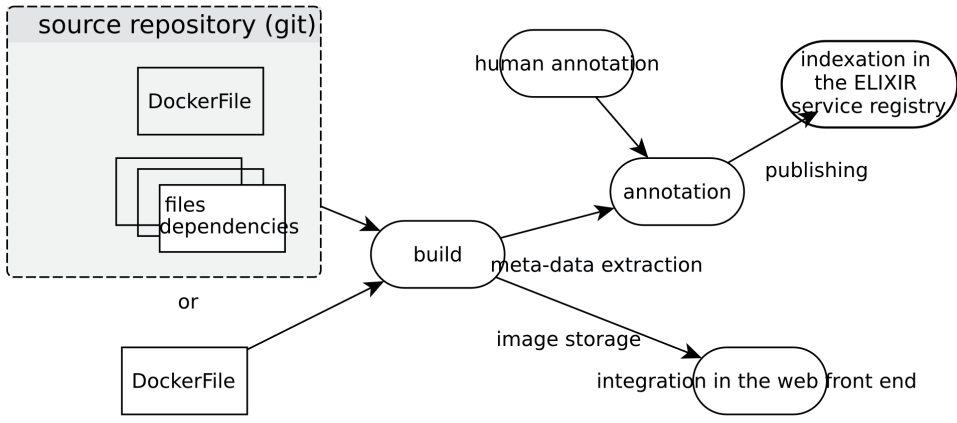


Figure 2. The BioShaDock Docker container processing steps.

update or via a link (URL with an API Key), shown in the web interface when the user is the owner of the tool (created it). The creation of a tag on the image uses the same link mechanism. Such a link can be used directly (copy/paste in the browser) or via external tools or hooks (GitHub web hooks for example). The API also provides the possibility to trigger it manually, or to tag a container (i.e. set a version).

The Docker images, once built and stored in BioShaDock, can be registered in the ELIXIR registry (using some LABEL metadata in the Dockerfile). It is also possible to add a link to an existing ELIXIR registry entry. By linking its contents to and from the ELIXIR registry, BioShaDock enables the discovery of Docker images from a more generic system where users might look for a given software without specifically searching for container solutions. It hence maximizes the visibility of its images and contributes to better software dissemination.

Search and execution

Listing 1. An example of Docker image command line invocation using BioShaDock. After an automatic download, the container is executed. Here, the program BWA is called by default.

```
sudo docker run docker-registry.genouest.org/
bioinfo/\bwa
Unable to find image \
'docker-registry.genouest.org/bioinfo/bwa:
```

```
latest'\locally
latest: Pulling from bioinfo/bwa
[...]
Status: Downloaded newer image for \
docker-registry.genouest.org/bioinfo/bwa:latest
Program: bwa (alignment via Burrows-Wheeler \
transformation)
Version: 0.7.5a-r405
[...]
```

The images provided by BioShaDock can be executed in various ways (Figure 3):

The Docker repository acts as a platform that facilitates the dissemination of bioinformatics tools by providing ready to use Docker images.

- on a personal computer with a Linux system (Windows and Mac are supported with the Docker Toolbox), in a command line (Listing 1), directly using Docker¹⁴;
- on a cluster integrating a Docker scheduler front-end like GO-DOCKER (v1.0)¹⁶;
- in any software implementing the CWL (Common Workflow Language) specification (draft 3)^{19,20} such as Arvados²¹ or Rabix (v0.6.5)²²;

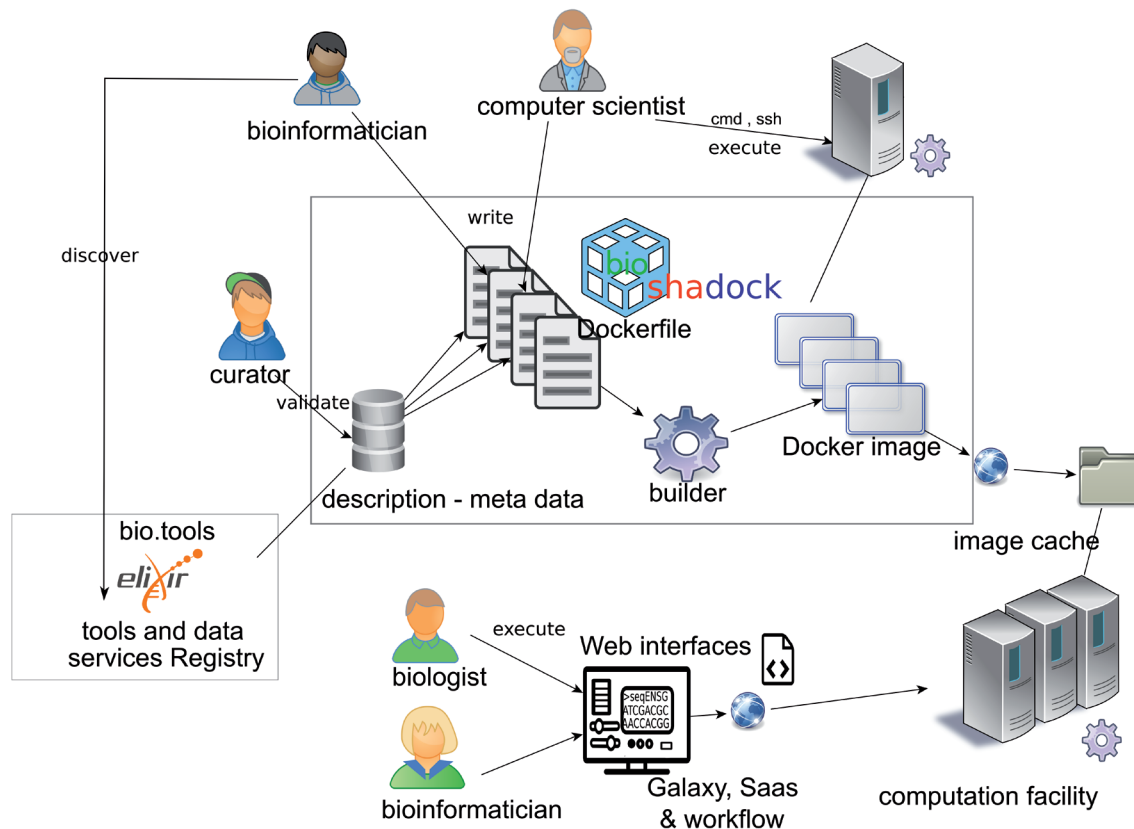


Figure 3. The BioShaDock use cases. The Docker repository acts as a platform that facilitates the dissemination of bioinformatics tools by providing ready to use Docker images.

- in the D⁴ workflow portal²³ (v0.6);
- in the Galaxy environment⁶ (v15.10);
- in the cloud of the French Institute of Bioinformatics with the help of the Docker virtual machine image²⁴.

As an illustration, we created a set of Galaxy tool descriptors based on Docker images stored by BioShaDock²⁵ available in our Toolshed²⁶. Thus, the stacks RADSeq pipeline²⁷ is available as a Galaxy tool xml descriptor²⁸ that calls a container stored in BioShaDock²⁹.

Implementation

Listing 2. A container 'Dockerfile' that defines the automated image build process. The LABEL instructions represent metadata.

```

LABEL name="Emboss"
LABEL homepage="http://emboss.sourceforge.net/"
LABEL resourceType="Tool"
LABEL interfaceType="Command line"
LABEL description="The European Molecular \
  Biology Open Software Suite"
LABEL topic="Data processing and validation"
#EDAM operation
LABEL functionName="Sequence processing"
FROM biodockr/biodocker:latest
USER root
# Install EMBOSS package
RUN apt-get update && \
  apt-get install -y \
    emboss=6.6.0-1 && \
  apt-get clean && \
  apt-get purge && \
  rm -rf /var/lib/apt/lists/* /tmp/* /var/tmp/*

USER biodocker
WORKDIR /data
CMD ["embosdata"]
MAINTAINER Adam Smith <asmithswx@cnrs.fr>

```

BioShaDock is a web application written in python (>=2.7). It manages the container's build and metadata. It is also in charge of authenticating the user against a local Docker registry and authorizing the user to push or pull a container according to their role (admin, editor, etc.) or rights. A user can give other users access to their repository for collaborative work in the edition page of the tool. Collaborators can have read only (for private repositories) or read/write access to the tool. The backend is based on a local instance of a Docker registry.

A script extracts the metadata written by the image's maintainer (Listing 2).

Listing 3. An XML container metadata description generated from the LABEL instructions by BioShaDock and used to publish the container metadata in bio.tools, the ELIXIR registry.

```

<?xml version="1.0" encoding="UTF-8"?>
<resources xmlns="http://bio.tools">
  <resource>
    <name>ngs_multi_vendor_read_corrector</name>
    <homepage>http://resourcename.org</homepage>
    <resourceType>Tool</resourceType>
    <interface>
      <interfaceType>Command line</interfaceType>
    </interface>
    <description>
      software analysis package specially
      developed for the needs of the molecular
      biology user community
    </description>
    <topic uri="http://edamontology.org/topic_0220">
      Data processing and validation
    </topic>
    <function>
      <functionName
        uri="http://edamontology.org/operation_2446">
        Sequence processing
      </functionName>
    </function>
    <contact>
      <contactEmail>
        asmithswx@cnrs.fr
      </contactEmail>
    </contact>
  </resource>
</resources>

```

Then, an integrated REST python client (v1.0) manages the container indexation in bio.tools (Listing 3). The first version of the registry integrates 80 Docker images that are versioned and can be re-built when the sources are updated. A REST API enables programmatic interaction with the server. For example, it can be used by external tools to extract the list of available images for job submissions. GO-DOCKER (v1.0) and the D⁴ workflow portal (v0.6) integrate this feature. The access to the images is public. To ensure the quality of available images, BioShaDock manages the authentication and ACL (access control list) to restrict the creation and update of its images to identified trustful contributors. The current implementation (v1.0) enables authentication using LDAP, Google or GitHub.

Discussion

The aim of BioShaDock is to contribute to the aggregation and standardization of bioinformatic tools and utilities. Maintaining ready to use validated and versioned software is key in ensuring the reproducibility needed in an open science approach.

Thereby, the creation of a collection of tools embedded in Docker containers, as provided by BioShaDock, is a pragmatic solution to this major bottleneck.

A number of other projects also focus on the provision of bioinformatic Docker images. BioDocker³⁰ is a community based initiative to encourage the use of Docker images in bioinformatics. A GitHub repository stores a list of Dockerfiles that define the construction of images for the corresponding bioinformatic tool, with an open yet controlled contribution mechanism. Bioboxes³¹ is an open source project that defines guidelines to build bioinformatic tool images using compatible interfaces for images which perform the same task, independent of the underlying tool, hence favoring interoperability between tools. It is therefore, among other characteristics, very well suited to automate tool and pipeline benchmarks. It has been applied to the assessment of different types of NGS data processing methods that concern assembly software as well as metagenomics tool. Dockstore³² is an open platform that enables the registration of Docker images described using CWL. It integrates with a number of external services for source code and image hosting, and focuses on the provision of images that can be integrated in CWL-ready environments. BioShaDock shares with these existing efforts the use of Docker as a container technology to facilitate the distribution and integration of bioinformatic tools. However, none of these systems are designed to provide local image building and storage options. Furthermore, we believe the integration of BioShaDock with external domain-centric and platform-agnostic registries such as the ELIXIR registry will significantly raise the visibility of both the images provided and the container technology itself to the community of bioinformatic tool users. Because the files that describe the image building process (Dockerfiles) are usually freely available online, the interoperability issues between Docker registry initiatives are potentially very limited.

Conclusions

Computer scientists and bioinformaticians can more easily disseminate their programs and find potential users using a dedicated domain-centric Docker registry. There is a wide range of perspective uses for container registries in bioinformatics: repositories managed at a community level, based on tools embedded in containers, promote the ability to exchange and replicate data analyses.

In addition, the association between workflow models, data references and containerized tools could lead to the creation of interoperable and ready to use analysis components and pipeline collections maintained by many contributors. The development of such specifications is already in progress as illustrated by the CWL (Common Workflow Language)²⁰ and the A-SCDFM (Autonomous Semi-Concrete Data Flow Model)³³ portable workflow formats that are natively compatible with containers. In this case, the integration of programs in a container registry like BioShaDock and the formalization of the data processing following one of these new portable workflow specifications could simplify the creation of

reproducible benchmarks, teaching material, demos and the production of use case prototypes. It could also be used by article reviewers to quickly evaluate a software.

The spread of container usage in the bioinformatics community and their indexing in repositories can be a solution to capture and share a large collection of data analysis methods. A wide set of bioinformatics components available on demand could induce better data analysis by simplifying tests and benchmarks.

Software availability

Server

- BioShaDock registry: <https://docker-ui.genouest.org>
- BioShaDock home page: <http://bioshadock.genouest.org>

Source code

- BioShaDock client and tools: https://github.com/fjrmoreews/bioshadock_client
- BioShaDock local server: <https://bitbucket.org/osallou/bioshadock>
- Archived source code at the time of publication (client): <https://zenodo.org/record/3458834>
- Archived source code at the time of publication (server): <https://zenodo.org/record/3458735>

License

Apache 2.0

Author contributions

FM and OS conceived the software and developed the web interface and the build system. HM participated to the meta-data publishing feature design. YLB and CM designed some of the first Dockerfile and integrated Docker images in our Galaxy toolshed. OC and CB managed the deployment and infrastructure availability. All authors helped prepare the manuscript.

Competing interests

No competing interests were disclosed.

Grant information

Funding was provided from the Western France e-science project supported by Brittany and Pays de la Loire regions (e-Biogenouest/052012).

I confirm that the funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgements

We would like to thank the IFB/Genouest platform for hosting the software and the use of its cluster to build the images. We thank the IFB-CORE team, especially Marie Grosjean and Sandrine Perrin, for creating containers and for supporting the deployment of the registry coupled with the IFB cloud facility. Finally, we also thank the BioDocker core team, Felipe da Veiga Leprevost, Yasset Perez-Riverol and Saulo Alves Aflitos for collaborative efforts.

Supplementary material

BioShaDock API documentation:

<http://www.genouest.org/api/bioshadock-api>

References

1. Woelfle M, Olliaro P, Todd MH: **Open science is a research accelerator.** *Nat Chem.* 2011; **3**(10): 745–748.
[PubMed Abstract](#) | [Publisher Full Text](#)
2. Stajich JE, Lapp H: **Open source tools and toolkits for bioinformatics: significance, and where are we?** *Brief Bioinform.* 2006; **7**(3): 287–296.
[PubMed Abstract](#) | [Publisher Full Text](#)
3. Ison J, Rapacki K, Ménager H, *et al.*: **Tools and data services registry: a community effort to document bioinformatics resources.** *Nucleic Acids Res.* 2015; pii: gkv1116.
[PubMed Abstract](#) | [Publisher Full Text](#)
4. Connor BO, Kartashov A, Yuen D, *et al.*: **ELIXIR Tools and Data Services Registry.** 2015.
[Reference Source](#)
5. Goecks J, Nekrutenko A, Taylor J, *et al.*: **Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences.** *Genome Biol.* 2010; **11**(8): R86.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
6. Aranguren ME: **Merging OpenLifeData with SADI services using Galaxy and Docker.** *BioRxiv, Cold Spring Harbor Labs.* 2015.
[Publisher Full Text](#)
7. Blankenberg D, Von Kuster G, Bouvier E, *et al.*: **Dissemination of scientific software with Galaxy ToolShed.** *Genome Biol.* 2014; **15**(2): 403.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
8. Lawlor B, Walsh P: **Engineering bioinformatics: building reliability, performance and productivity into bioinformatics software.** *Bioengineered.* 2015; **6**(4): 193–203.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
9. Prins P, de Ligt J, Tarasov A, *et al.*: **Toward effective software solutions for big biology.** *Nat Biotechnol.* 2015; **33**(7): 686–687.
[PubMed Abstract](#) | [Publisher Full Text](#)
10. Van Gorp P, Mazanek S: **SHARE: a web portal for creating and sharing executable research papers.** *Procedia Comput Sci.* 2011; **4**: 589–597.
[Publisher Full Text](#)
11. Granger B, Avila D, Perez F, *et al.*: **Jupyter: Open source, interactive data science and scientific computing across over 40 programming languages.** 2015.
[Reference Source](#)
12. Kanterakis A, Kuiper J, Potamias G, *et al.*: **PyPedia: using the wiki paradigm as crowd sourcing environment for bioinformatics protocols.** *Source Code Biol Med.* 2015; **10**(1): 14.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
13. Merkel D: **Docker: Lightweight Linux containers for consistent development and deployment.** *Linux J.* 2014; (239).
[Reference Source](#)
14. **Docker.** 2013; [Online; accessed 16-Nov-2015].
[Reference Source](#)
15. google Inc: **Google Container Engine.** 2015.
[Reference Source](#)
16. Sallou O, Monjeaud C: **GO-Docker: Batch scheduling with containers.** *IEEE Cluster 2015.* 2015.
[Reference Source](#)
17. Di Tommaso P, Palumbo E, Chatzou M, *et al.*: **The impact of Docker containers on the performance of genomic pipelines.** *PeerJ.* 2015; **3**: e1273.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
18. Ison J, Kalas M, Jonassen I, *et al.*: **EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats.** *Bioinformatics.* 2013; **29**(10): 1325–1332.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
19. Peter A, Nebojsa T, Stian SR, *et al.*: **Beyond Galaxy: portable workflows and tool definitions with the CWL.** In *Galaxy Community Conference 2015.* Norwich, United Kingdom, 2015.
[Reference Source](#)
20. Amstutz P, Chilton J, Crusoe MR, *et al.*: **Common Workflow Language.** 2015.
[Reference Source](#)
21. **Arvados.** 2015.
[Reference Source](#)
22. **Rabix.** 2015.
[Reference Source](#)
23. Francois M: **D4 Workflow Portal.** 2015.
[Reference Source](#)
24. IFB cloud: **The academic cloud of the French Institute of Bioinformatics.** Online; accessed 2015-09-24.
[Reference Source](#)
25. Moreews F, Sallou O, Bras YL, *et al.*: **A curated Domain centric shared Docker registry linked to the Galaxy toolshed.** In *Galaxy Community Conference 2015.* Norwich, United Kingdom, 2015.
[Reference Source](#)
26. Bras YL, Monjeaud C: **GUGGO Galaxy ToolShed.** 2014. [Online; accessed 05-Nov-2015].
[Reference Source](#)
27. Catchen J, Amores A, Hohenlohe P, *et al.*: **STACKS, a software pipeline for building loci from short-read sequence.** 2010. [Online; accessed 02-Dec-2015].
[Reference Source](#)
28. Bras YL, Monjeaud C: **STACKS pipeline, galaxy tool descriptor.** 2010. [Online; accessed 02-Dec-2015].
[Reference Source](#)
29. Bras YL, Monjeaud C: **STACKS pipeline, docker container.** 2010. [Online; accessed 02-Dec-2015].
[Reference Source](#)
30. **BioDocker.** 2015.
[Reference Source](#)
31. Belmann P, Dröge J, Brørges A, *et al.*: **Bioboxes: standardised containers for interchangeable bioinformatics software.** *Gigascience.* 2015; **4**: 47.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
32. Connor BO, Kartashov A, Yuen D, *et al.*: **DockStore.** 2015.
[Reference Source](#)
33. Moreews F: **Design and share data analysis workflows. Application to bioinformatics intensive treatments.** Thesis, université de rennes 1. 2015.
34. Francois M, Olivier S: **BioShaDock client.** *Zenodo.* 2015.
[Data Source](#)
35. Olivier S, Francois M: **BioShaDock server.** *Zenodo.* 2015.
[Data Source](#)