

Grant agreement no. 675451

CompBioMed**Research and Innovation Action**

H2020-EINFRA-2015-1

Topic: Centres of Excellence for Computing Applications

D6.5 Report on the Use of Commodity HPC Infrastructures

Work Package: 6

Due date of deliverable: Month 24

Actual submission date: 19 / October / 2018

Start date of project: 01 / October / 2016 Duration: 36 months

Lead beneficiary for this deliverable: ACE

Contributors: ACE

Project co-funded by the European Commission within the H2020 Programme (2014-2020)		
Dissemination Level		
PU	Public	YES
CO	Confidential, only for members of the consortium (including the Commission Services)	
CI	Classified, as referred to in Commission Decision 2001/844/EC	

Disclaimer

This document's contents are not intended to replace consultation of any applicable legal sources or the necessary advice of a legal expert, where appropriate. All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user, therefore, uses the information at its sole risk and liability. For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors' view.



Table of Contents

1	Version Log	3
2	Contributors	3
3	Definition and Acronyms	4
4	Introduction	4
5	Cloud based compute services in CompBioMed	5
5.1	AceCloud: Molecular Dynamics in the Cloud	5
5.2	DNAexus: a cloud-based Platform-as-a-Service	6
5.3	On-demand biomedical compute services	6
6	HPC and Cloud Platforms in CompBioMed	8
6.1	Metrocubo - Benchside machines with HPC power	8
6.2	CompBioMed HPC system	9
6.3	SURFsara HPC cloud system	9
7	Project Activities	9
7.1	Virtualization: a tool for achieving computational reproducibility	9
7.2	Container technologies for biomedical applications	11
7.3	Porting Consortium Applications to PlayMolecule: alpha porting	13
7.4	Security and Privacy: Containers and Cloud	14
8	Conclusions	15
9	Bibliography / References	16

List of Tables and Figures

Figure 1: Print screen of the AceCloud AWS Marketplace product page. Prices are blurred	5
Figure 2: The latest version of a Metrocubo workstation	8
Figure 3: Print screen of the Acellera organization profile and channel page on Anaconda Cloud (https://anaconda.org/acellera)	10
Figure 4: Print screen of Acellera's HTMD package page on Anaconda Cloud (https://anaconda.org/acellera/htmd)	11
Figure 5: Container comparison (taken from [Kurtzer 2017])	12
Figure 6: Print screen of the homepage of the PlayMolecule platform (www.playmolecule.com)	14
Table 1. List of CompBioMed applications which used container technologies.	13
Table 2: Qualitative analysis of trust in security level of computing resources used in industry, non-profit and hospital. Secure refers to the users agreement that using the corresponding resources is safe enough for the data they deal with. Data used were divided in 2 classes: scientific data (mainly for molecular simulations) and clinical data (mainly related with diagnostics, analysis within or not a clinical trial).	14



1 Version Log

Version	Date	Released by	Nature of Change
V0.1	17/07/2018	João M. Damas	Preliminary Draft
V1.0	04/09/2018	João M. Damas	Initial Draft for CompBiomed Internal Review
V1.1	09/09/2018	João M. Damas	Final Draft for CompBiomed Internal Review
V1.2	24/09/2018	João M. Damas	Final Draft for CompBioMed leaders final Review
V1.3	01/10/2018	João M. Damas	Draft after CompBioMed leaders review
V1.4	15/10/2018	Peter V. Coveney	Final draft

2 Contributors

Name	Institution	Role
João M. Damas	Acellera	Main author
Gerard Martínez-Rosell	Acellera	Author
Marco Verdicchio	SURFsara	Author
Raimondas Galvelis	Acellera	Contributor
Alberto Marzo	USFD	Reviewer
Erwan Raffin	BULL	Reviewer
Marco Stijnen	LTG	Reviewer
Tim van den Boom	LTG	Reviewer
Peter Coveney	UCL	Reviewer
Emily Lumley	UCL	Reviewer

3 Definition and Acronyms

Acronyms	Definitions
HPC	High-Performance Computer
OS	Operating System
MD	Molecular Dynamics
GPU	Graphical Processing Unit
AWS	Amazon Web Services
ND	Not determined

4 Introduction

The work described in this deliverable **D6.5: Report on the Use of Commodity HPC Infrastructures** is related to the activities developed by Acellera under task **Task 6.3: Adaptation to Commodity HPC Infrastructures**, a deep-track task that aims to bring Biomedical Applications from HPC centres to industrial and clinical settings. The objectives of this deliverable are to report on these developments, namely the implementation of technology to facilitate the porting of such applications to commodity HPC infrastructures. This approach not only ensures that data protection and ethics rules be enforced, it also means that important industrial or clinical simulations do not have to contend with work done by other users.

The world-wide push for Cloud Computing by many multinational companies working in the area of HPC has been changing the landscape in the way they purchase and manage their computational resources. CompBioMed has also adapted to this, and we have expanded the initial definition of “commodity HPC infrastructures” (as stated in the Grant Agreement) to include Cloud Computing, as it has become a “commodity” in contemporary HPC standards. Today’s generic HPC environments, characterized by highly parallel and memory efficient applications that run on many cores, occupy in fact only the lowest layers of the vertically integrated application stacks that the CompBioMed Centre of Excellence (CoE) will develop. CompBioMed users require not only high computing power, but also flexible and secure access to the resources which cannot always be fulfilled by using traditional HPC infrastructure.

In this report we describe the activities carried out within the CompBioMed project, around the porting to commodity HPC infrastructure of biomedical workflows and applications. In Section 5, we introduce the tools and services available within the consortium to deploy and run cloud-based biomedical services. Next, we describe some of the infrastructures available to CompBioMed users for the development of such compute services. Finally, we report on current efforts on the porting of applications within the consortium to commodity HPC infrastructures with a focus on virtualization schemes and how to address these challenges to construct industry-like cloud-ready environments.

It is important to note that, despite the delivery of this document in M24, many activities of Task 6.3 are still ongoing, as the duration of this task is from M12 - M36. The final section describing the current tasks and future perspectives, accounts for this.

5 Cloud based compute services in CompBioMed

The advent and popularization of cloud computing, specifically after Amazon Web Services (AWS) released its Elastic Compute Cloud product in 2006, has changed the worldwide landscape of HPC, in particular regarding business decisions (e.g. buying computing resources for long term use vs. requesting computing resources on-demand based on necessity). In 2010, GPU instances became available on AWS, which was of particular interest to the research and modelling community. Recently, with the availability of resources greatly increasing (and improving in terms of hardware quality) and prices also decreasing (both through resources becoming cheaper and due to spot pricing model), cloud computing is gaining increasing traction and many tools and frameworks are currently available for the porting of HPC applications to cloud computing infrastructures.

In this section we describe the main tools available within the CompBioMed project, to develop and execute cloud based biomedical compute services.

5.1 AceCloud: Molecular Dynamics in the Cloud

AceCloud [Harvey 2015] is a software, developed by Acellera, that allows people to run MD simulations on the Cloud, namely under AWS. The only real extra steps necessary are to have an AWS account and subscribe to the AceCloud Marketplace product (Figure 1).

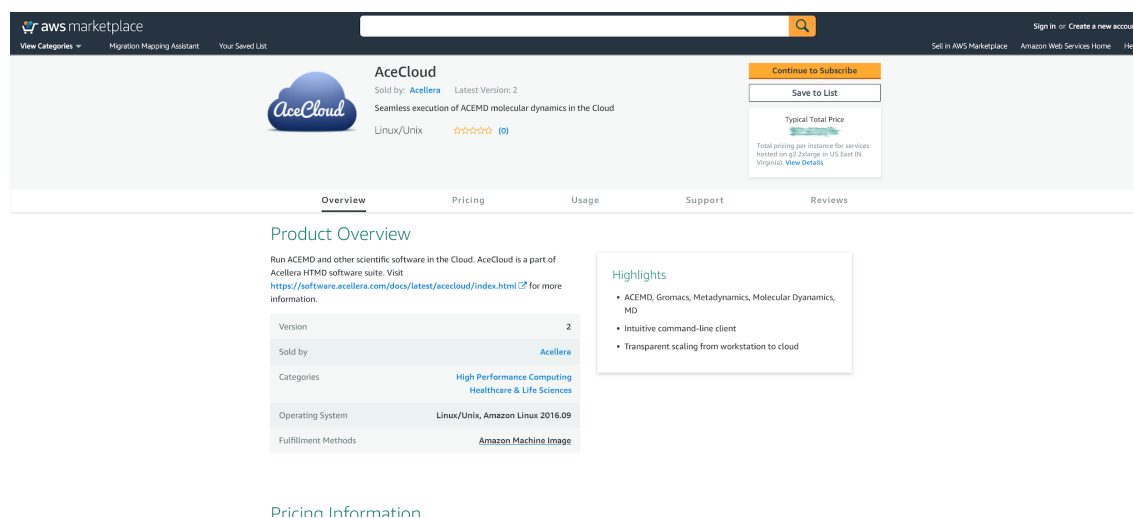


Figure 1: Print screen of the AceCloud AWS Marketplace product page. Prices are blurred

Many of the concerns about privacy and security regarding running MD simulations on the Cloud are mitigated by the Compliance certifications that AWS possesses (<https://aws.amazon.com/compliance/programs/>) and by the security protocols implemented. Ultimately, AceCloud is a stand-alone client that follows the highest standards of security, with the data sent and received being encrypted and transient (following state of the art encryption protocols), while keeping the simplicity of use for the end-user. Nonetheless, despite these security measures, we have identified that many companies dealing with sensitive, confidential

and private data have concerns over the data flowing in and out of their internal infrastructure and network (where security can be assured and verified more easily). Abiding by compliance programs and filing for their certification is a first step, but many companies may still have long security review processes that may hinder the immediate use of Cloud resources. The consortium has both core and associate partners working in this area and we will be working together to improve trust in security compliances for Cloud systems.

5.2 DNAnexus: a cloud-based Platform-as-a-Service

The DNAnexus platform, is a cloud-based Platform-as-a-Service (PaaS) optimized to address the challenges of security, scalability, and collaboration, for organizations that are pursuing genomic-based approaches to health, in the clinic and in the research lab. The platform supports the ingestion of any type of data and any type of Linux-based software (your own, commercial or open-source) for the analysis of said data. Developed by the CompBioMed associate partner DNAnexus, its main focus is on genomics but it also includes applications in the Computational Chemistry space. UCL and EnsembleMD have been working closely with DNAnexus and have a deployment of BAC running on their platform.

5.3 On-demand biomedical compute services

Description of the on-demand biomedical services accessible through the CompBioMed project:

- **InSilicoMRI:** InSilicoMRI provides a framework to predict the overheating of a medical device during an MRI scan. This software allows to evaluate the radiofrequency (RF) safety analysis of a passive device exposed to a 3T MRI birdcage coil field following the directives of ASTM F2182 standard. The simulation calculates the EM fields, SAR, and thermal heating after 900s of RF exposure. This application use Microsoft Azure cloud services to run the simulation.

<http://www.compbiomed.eu/services/software-hub/compbiomed-software-insilicomri/>

- **Living Heart Human Model:** The SIMULIA Living Heart Human Model is a high-fidelity multiphysics model of a healthy, 4-chamber adult human heart and proximal vasculature. The dynamic response of the Living Heart is governed by realistic electrical, structural, and fluid (blood) flow physics. With this model, medical professionals, researchers, and device manufacturers will be able to rapidly conduct virtual experiments in a highly realistic 3D environment. The Living Heart can readily be used to study cardiac defects or diseased states and explore treatment options by modifying its geometry, loading, or electromechanical properties. In addition, medical devices can be inserted into the model to study their influence on cardiac function, validate their efficacy, and predict their reliability under a wide range of operating conditions.

<http://www.compbiomed.eu/services/software-hub/compbiomed-software-living-heart-human-model/>



- **VisualGec:** Visual GEC is a software tool for designing engineered cells and simulating biochemical interactions. The Genetic Engineering of Cells (GEC) software, developed by the Biological Computation team at Microsoft Research (Cambridge, UK), is a modelling tool that can be used to design and simulate synthetic genetic circuits. At the core is a domain-specific programming language for biochemical systems (LBS), originally developed at the University of Edinburgh. The tool supports stochastic and deterministic simulation of the temporal dynamics of chemical reaction networks, but also spatio-temporal dynamics via reaction-diffusion equations. Parameter inference can also be performed using Metropolis-Hastings Markov chain Monte Carlo with time-series data.

<http://www.compbioimed.eu/services/software-hub/compbioimed-software-visual-gec/>

- **Virtual Assay:** The Virtual Assay software provides a framework to run in silico drug trials in populations of human cardiac cell models for predictions of drug safety and efficacy. Virtual Assay starts with well-understood human cellular biology models and modulates the variables to generate a range, or population, of models, which will respond differently to the same inputs. These populations are then calibrated against experimental data, retaining only those models in Calibrated Model Populations range with experimental observations. Once calibrated, these populations can be used to analyse the effects of different pharmaceutical agents on cellular response at the population level.

<http://www.compbioimed.eu/services/software-hub/compbioimed-software-virtual-assay/>

- **CT2S:** Computer Tomography to Strength (CT2S) is an on-line service developed by the team of Prof Marco Viceconti at the Insigneo Institute at the University of Sheffield (UK), which allows the prediction of the biomechanical strength of a patient's bone from a clinical CT scan of that bone. The service operates by creating a patient-specific finite element model of the bone, using a state-of-the-art image-processing pipeline. This very precise model of the patient's anatomy is then examined under a range of highly realistic simulated loading conditions, including walking, running, stair-climbing and falling, and the fracture load is computed in each case. Data summarising the identified fracture strength is returned to the user. The solution is currently exposed as a service, accessible through a web interface; the backend HPC system currently in use is USFD's own ShARC. The service is currently provided at cost, with a significant discount for non-sponsored clinical studies. USFD is currently exploring the best marketing strategy.

<http://www.compbioimed.eu/services/software-hub/compbioimed-software-ct2s/>

6 HPC and Cloud Platforms in CompBioMed

Through its wide ranging and numerous partners, CompBioMed users can get access to specialized hardware to set up and run their simulations. In this section we provide a description of the main CompBioMed infrastructures available for the development and execution of cloud-like applications.

6.1 Metrocubo - Benchside machines with HPC power

Acellera has been the pioneer in the development of MD engines to run on Graphical Processing Units (GPUs), namely ACEMD [Harvey 2009], when all the popular MD engines were still only capable of running on CPUs. The introduction of ACEMD allows it to be run on a desktop machine with 4 GPUs that would be equivalent in computing power to a small CPU cluster. Of course, not all codes within computational biomedicine are amenable to GPU acceleration in any simple manner. However, many of the top supercomputers in the world include accelerators such as GPUs. The porting of executables that run on multi-core architectures to those with accelerators is likely to be an inevitable requirement for applications to run at the exascale.

Despite the big advantage of GPUs for certain type of applications, limitations are evident: the price of GPUs, despite being driven by the gaming community (and in a pre-cryptocurrency-mining age), are still fairly high. In the face of such costs, Acellera has been obliged to develop and release Metrocubo, a desktop machine (workstation) with 4 GPUs, which may also be incorporated within racks in computing centres (Figure 2).



Figure 2: The latest version of a Metrocubo workstation

Metrocubo had all the necessary hardware and software requirements for the applications expected to be run on it (initially, ACEMD). The release of Metrocubo resulted in a reduction of time and effort in reaching the end-user; they can run MD simulations as soon as they connect to the machine and launch the operating system (OS). Regardless of this advancement, issues were exposed through time and evolution of the market. The necessity of seamless compatibility with other emerging hardware and infrastructures (the arrival and expansion of commodity HPC infrastructures) has driven us to consider and develop new solutions to deploy and update Acellera software. This became more evident when we developed and ported HTMD [Doerr 2016], a programming environment for molecular modelling and simulation developed by Acellera, to Python: a python-based software brought many challenges in terms of deployment, due to its intrinsic modular nature and constant development at a high pace,

making version/deployment control and releasing schemes much more important for the success of the applications in such HPC environments.

6.2 CompBioMed HPC system

In addition to the PaaS identified, the CompBioMed Centre of Excellence provides access to a wide range of computational resources to the consortium members. These include a European Tier 0 (supra-national European HPC resource, e.g. Piz Daint), Tier 1 (national HPC resource, e.g. ARCHER and Cartesius) and Tier 2 (regional HPC resource, e.g. Cirrus, Lisa) supercomputers. Most of these systems are currently equipped with GPUs accelerators and allow the execution of containerized applications (i.e. Lisa and Cartesius at SURFsara, along with Cirrus at EPCC).

6.3 SURFsara HPC cloud system

SURFsara offers the HPC Cloud computing facility as an Infrastructure as a Service (IaaS) platform. IaaS allows users to get their own (virtual) environment, storage and networking resources, where users can deploy, configure and run arbitrary software (including your operating system of choice). SURFsara has implemented the Cloud service relying on the OpenNebula software [Moreno-Vozmediano 2012]. Users interact with the HPC Cloud via the web interface that OpenNebula offers.

7 Project Activities

7.1 Virtualization: a tool for achieving computational reproducibility

One of the main problems with deploying software in arbitrary hardware infrastructures is system heterogeneity (for example: OS, drivers, libraries). With Metrocubo (see section 5.1.1), this issue is circumvented because there is total control over system installation: full compatibility with the software can be guaranteed before shipping. However, for other systems, and for system updates on Metrocubo machines, this could not be guaranteed, so a solution was necessary.

The initial solution employed was the use of virtual environments for a “soft” independence of the software from the system. In particular, we found that Anaconda [Anaconda 2018], a free and open source distribution of the Python and R programming languages for data science and machine learning related applications, was a good choice. The widespread use of Python programming language within the scientific community and the fact that HTMD (written in Python) was one of the main concerns in terms of compatibility, this was a good choice. In fact, Anaconda allows the creation of multiple virtual environments with different versions of Python, which enables one to run, for example, HTMD in Python 3.6, without compromising the function of the OS that may rely on other versions of Python.

Acellera has created its own channel under Anaconda Cloud (Figure 3), which contains most of the company software stack, including dependencies. Automated tools for the release of software enables direct creation of the packages on the Anaconda Cloud channel within a few minutes. This allows fast and reliable deployment of the software in any type of infrastructure.



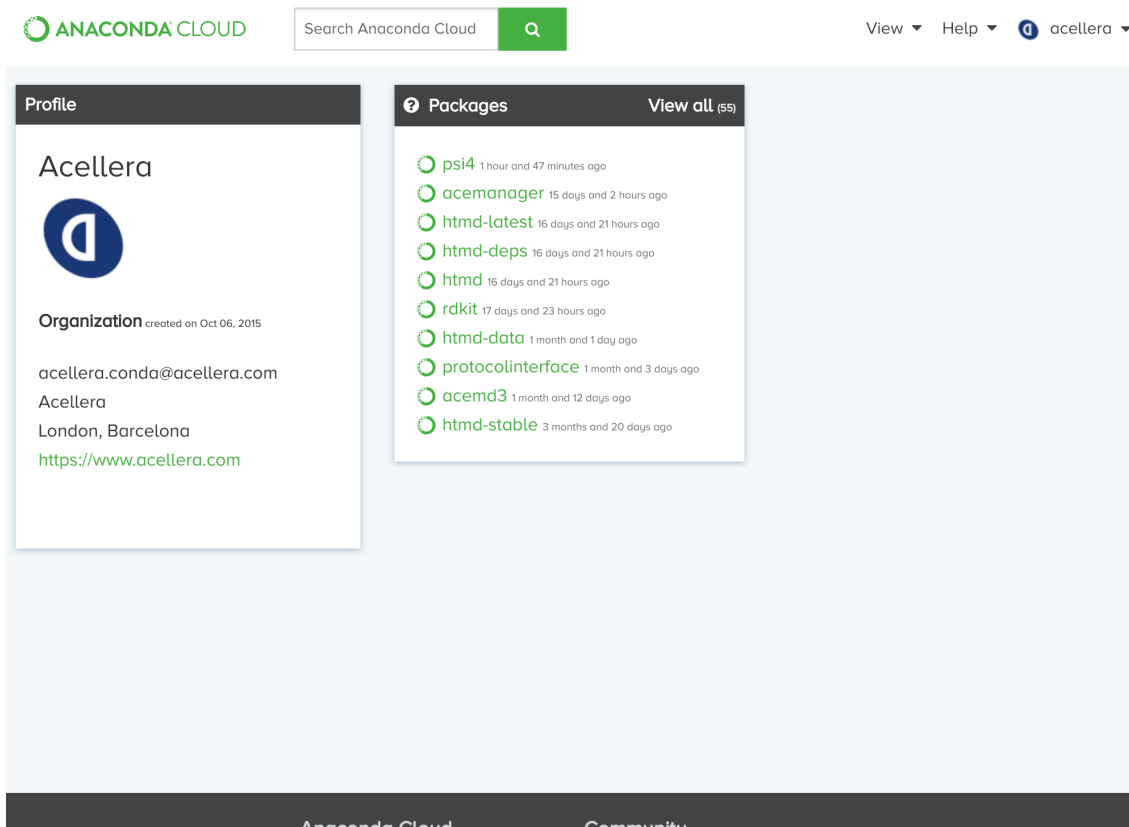


Figure 3: Print screen of the Acellera organization profile and channel page on Anaconda Cloud (<https://anaconda.org/acellera>)

A good example of a package distributed in Acellera’s channel is HTMD (Figure 4). HTMD is one of the most complex software products present in the channel in terms of the quantity of dependencies it relies upon. The dependency stack of HTMD comes from the modular nature of Python: many functionalities already exist in several modules (libraries) that are available for use in development of more complex software. Many of these modules are normally under active development and many new versions are regularly released. This means that maintaining the dependency stack in a healthy and functioning state is a requirement that Anaconda positively assists with. Some of its functionalities include the “environment solving” function, which determines the packages to install (or update) based on all the restrictions in the different versions and dependency requirements.

It should be explicitly understood that virtualisation brings with it one very important benefit, namely that simulations can then be run reproducibly across multiple different platforms. This is of fundamental importance in the context of biomedical computing aimed at supporting clinical outcomes.

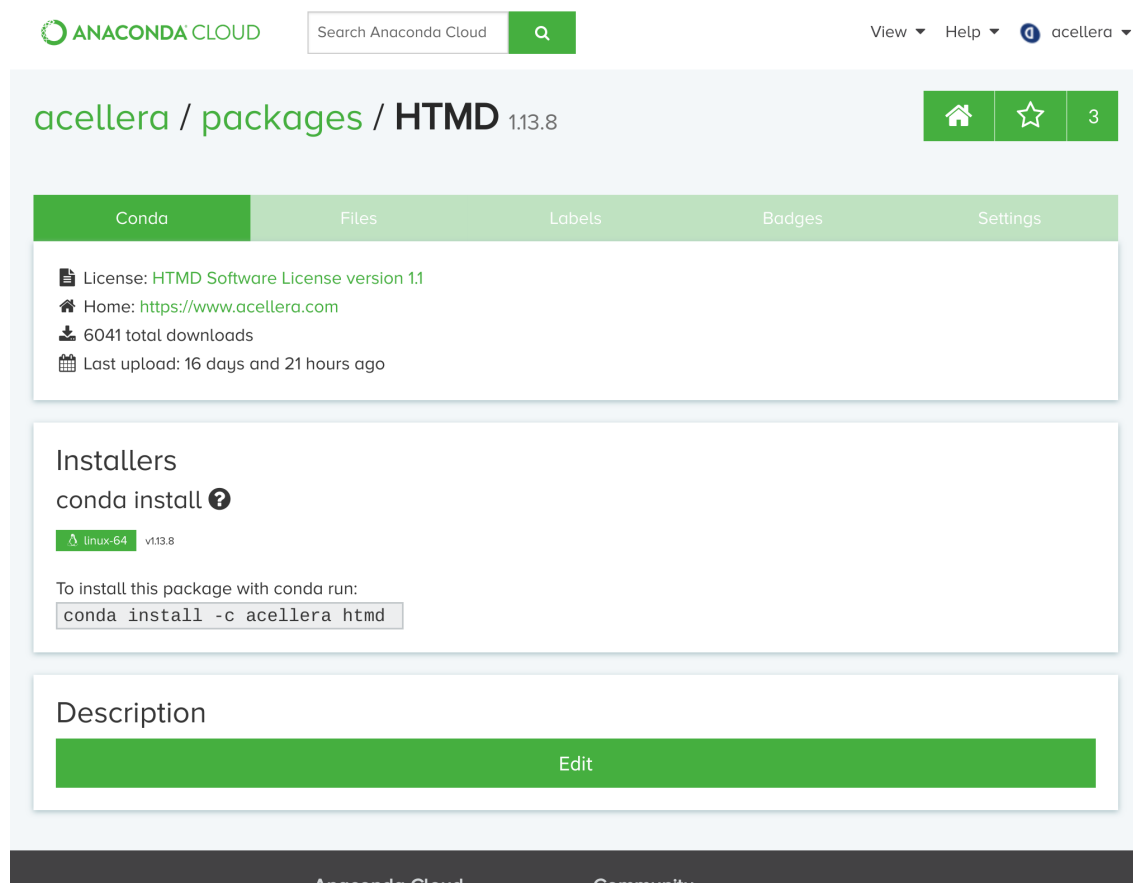


Figure 4: Print screen of Acellera's HTMD package page on Anaconda Cloud (<https://anaconda.org/acellera/htmd>)

Regardless of its merits, Anaconda still carries drawbacks. For example, in our use of Anaconda we have noticed that its update functionality is not stable in certain scenarios, either through incorrectly solving the environment or by updating the dependencies incorrectly. The latter can happen when dependencies contain C or other types of binary libraries. For production deployments in industrial context, Acellera has developed a small tool called acemanager, which allows a better management of the Anaconda environment when it comes to updating the software stack. Nonetheless, Anaconda is a virtual environment system and its “soft” isolation hinders some of the other objectives of this deliverable: full reproducibility in commodity HPC infrastructures.

7.2 Container technologies for biomedical applications

The virtual environments provided by Anaconda referred to in the previous section is not the only type of virtualization available. Virtualization can be divided mainly into three types: virtual environments, OS-level virtualization (container), and hardware virtualization. Each of these types has advantages and disadvantages.

One of the main disadvantages of virtual environments concerns the system libraries, which, unlike the other virtualization types, are shared with the host system. This may cause issues in the compatibility and portability of certain applications. Both OS-level and hardware virtualization provided the necessary level of isolation to carry on and implement.

Between OS-level and hardware virtualization, other features have been assessed. While having a private kernel would seem like the ideal situation (for complete system reproducibility), the reality is that it complicates the use of specific hardware, in particular GPUs, which are important for many applications. Furthermore, hardware virtualization can affect the performance of applications, which is undesired. As such, further work was carried out taking OS-level virtualization into consideration, i.e. containers.

There are many implementations of OS-level virtualization (containers) already in existence, with Docker [Docker 2018] currently being the industry standard. The numerous differences between these many implementations go beyond the scope of this deliverable. However, some of these differences between selected container implementations are shown in Figure 5.

Despite being the industry standard, Docker has security concerns that prevent its use in HPC infrastructures and in many industrial settings where HPC-like infrastructures are implemented (it may be viable in particular contexts, namely in small infrastructures with a tight system administrator control). In particular, the security concerns of Docker include access to the host filesystem and user escalated privileges (i.e. an unprivileged user being able to escalate to system administrator status).

	Singularity	Shifter	Charlie Cloud	Docker
Privilege model	SUID/UserNS	SUID	UserNS	Root Daemon
Supports current production Linux distros	Yes	Yes	No	No
Internal image build/bootstrap	Yes	No*	No*	No***
No privileged or trusted daemons	Yes	Yes	Yes	No
No additional network configurations	Yes	Yes	Yes	No
No additional hardware	Yes	Maybe	Yes	Maybe
Access to host filesystem	Yes	Yes	Yes	Yes**
Native support for GPU	Yes	No	No	No
Native support for InfiniBand	Yes	Yes	Yes	Yes
Native support for MPI	Yes	Yes	Yes	Yes
Works with all schedulers	Yes	No	Yes	No
Designed for general scientific use cases	Yes	Yes	No	No
Contained environment has correct perms	Yes	Yes	No	Yes
Containers are portable, unmodified by use	Yes	No	No	No
Trivial HPC install (one package, zero conf)	Yes	No	Yes	Yes
Admins can control and limit capabilities	Yes	Yes	No	No

In addition to the default Singularity container image, a standard file, Singularity supports numerous other formats described in the table. For each format (except directory) the suffix is necessary for Singularity to identify the image type.

*relies on Docker

**with security implications

***depends on upstream

<https://doi.org/10.1371/journal.pone.0177459.t001>

Figure 5: Container comparison (taken from [Kurtzer 2017])

Within the CompBioMed partners, we have identified Singularity [Kurtzer 2017] as the most suitable solution in terms of compatibility, portability, and security of applications across HPC-like infrastructures, including commodity HPC. By most suitable we consider a combination of its suitability for different infrastructures (academic and industrial, HPC and commodity HPC) and its current supported features, resulting in the greatest flexibility in the future. Under Singularity, it is also possible to run Docker containers, so compatibility can be assured to a certain extent, but we think that developing the containers themselves already on Singularity prevents problems that may occur down the road (example: accounting for GPU support). Finally, we have seen that performance inside Singularity containers is not affected. Of course, this may need further testing and benchmarking for a wider set of applications and resources.

The current implementations of containers are under intensive development. On the one hand, Docker developers will probably try to mitigate its security issues in future releases (an attempt was made when releasing version 1.10, for example). On the other hand, new features and new implementations may appear that may cause the current solution to be adapted to include the new features or to be replaced by another solution. Nonetheless, our plan is to carry on with the use of the container-based deployment of applications, while keeping the applications themselves implementation agnostic (i.e., not dependent on Singularity), in order to allow future changes to occur in the most seamless way possible.

Within the CompBioMed project, we are continuously working to extend the usage of container technologies within the biomedical community. Table 1 shows a summary of the codes, developed and maintained by the consortium, which are also available as containerized applications.

Table 1. List of CompBioMed applications which used container technologies.

Name	SoftwareHub Link	Container technology adopted
Alya	http://www.compbioMed.eu/services/software-hub/compbioMed-software-alya/	Singularity, Dockers, Shifter
BAC	http://www.compbioMed.eu/services/software-hub/compbioMed-software-bac/	Singularity, Dockers
Hemocell	http://www.compbioMed.eu/services/software-hub/compbioMed-software-hemocell/	Singularity, Dockers
HemeLB	http://www.compbioMed.eu/services/software-hub/compbioMed-software-hemelb/	Singularity, Dockers
Palabos	http://www.compbioMed.eu/services/software-hub/compbioMed-software-palabos/	Singularity
OpenBF	http://www.compbioMed.eu/services/software-hub/compbioMed-software-openbf/	Singularity

7.3 Porting Consortium Applications to PlayMolecule: alpha porting

PlayMolecule is a web-based repository for applications for molecular predictions and modelling (Figure 6). It not only hosts the applications but also allows users to execute them. It has been a playground for the implementation of container-based applications in its back-end with many applications created by Acellera and UPF already present there.

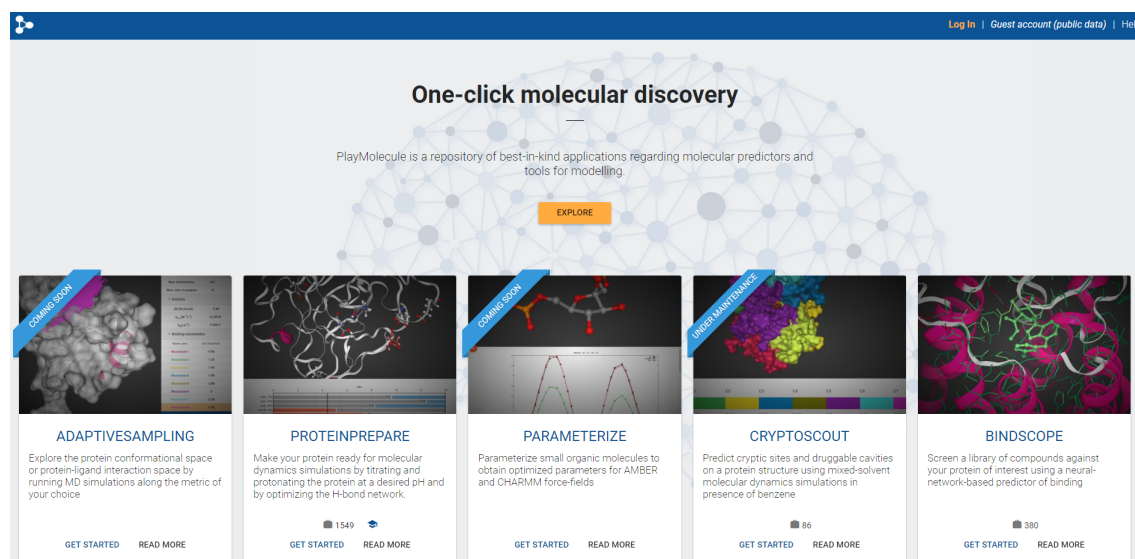


Figure 6: Print screen of the homepage of the PlayMolecule platform (www.playmolecule.com)

Within CompBioMed, WP6 is currently assessing the possibility of applying the lessons learned and development undertaken here to other applications within the consortium. Currently, the assessment of the readiness of the applications to be ported is being actioned through a questionnaire shared with all partners and associate partners. This is currently underway and this initial assessment will indicate the necessary activities for the alpha porting of a selected application to be performed, which should be concluded by M36.

7.4 Security and Privacy: Containers and Cloud

All the work performed was carried out under the best practices of security and privacy. Nonetheless, it is important for us to have the technology developed to follow internationally recognized security compliances. This is under the tasks to be developed in the M24-M36 period. Table 2 shows some qualitative analysis about data security in several different platforms.

Table 2: Qualitative analysis of trust in security level of computing resources used in industry, non-profit and hospital. Secure refers to the users agreement that using the corresponding resources is safe enough for the data they deal with. Data used were divided in 2 classes: scientific data (mainly for molecular simulations) and clinical data (mainly related with diagnostics, analysis within or not a clinical trial).

		Desk side	Private cloud	Public cloud or distributed computing resources	HPC
Industry scientific data	-	secure and used	secure and used	nd	non-sensitive data
Industry clinical data	-	secure and used	nd	nd	nd

Academy scientific data	-	secure used	and	secure used	and	secure used	and	secure used	and
Academy clinical data	-	secure used	and	nd		nd		nd	
Hospital scientific data	-	secure used	and	nd		secure used	and	nd	
Hospital clinical data	-	secure used	and	nd		nd		nd	

8 Conclusions

In conclusion, the work reported here is in line with the expectations for the deliverable. We have revised the task objectives according to business and market evolution and decisions taken in these sectors. In an initial approach, an Anaconda environment and channel was developed to address the reproducible deployment of applications on diverse infrastructures, including commodity HPC. Following this work, Singularity was identified as the most suitable solution for portable, compatible, and secure execution of applications under commodity HPC environments, and container-based applications were developed using this technology. As such, biomedical simulations and machine learning applications have been tested and continue to be tested under small clusters (commodity HPC infrastructures) under the chosen container-based approach.

The current activities include the porting of other applications within the consortium to commodity HPC using the aforementioned approach. Furthermore, security and privacy compliances are being worked on and strengthened to increase trust in the selected technology.

The partners of this task and WP6 are also collaborating with partners in WP2 and WP5, in order to make sure that the applications and technologies ready to deploy and be run by end users are compatible with the available compute power and machinery.

9 Bibliography / References

- Harvey M, Giupponi G, De Fabritiis G (2009) ACEMD: Accelerated molecular dynamics simulations in the microseconds timescale. J. Chem. Theory and Comput. 5, 1632 (2009). DOI: 10.1021/ct9000685
- Doerr S, Harvey MJ, Noé F, De Fabritiis G (2016) HTMD: High-throughput molecular dynamics for molecular discovery. J. Chem. Theory Comput. 12 (4), pp 1845–1852. DOI: 10.1021/acs.jctc.6b00049
- Harvey MJ, De Fabritiis G (2015) Acecloud: Molecular Dynamics Simulations in the Cloud. J. Chem. Inf. Model. 55 (5), 909. DOI: 10.1021/acs.jcim.5b00086
- Moreno-Vozmediano R, Montero RS, Llorente IM (2012) [IaaS Cloud Architecture: From Virtualized Datacenters to Federated Cloud Infrastructures](#), IEEE Computer, vol. 45, pp. 65-72. Web. <http://opennebula.org/>
- Anaconda Software Distribution. Computer software. Vers. 5-5.2.0. Anaconda, May 2018. Web. <https://anaconda.com>.
- Docker Community Edition. Computer software. Vers. 18.03.1-ce. Docker, April 2018. Web. <https://docker.com>.
- Kurtzer GM, Sochat V, Bauer MW (2017) Singularity: Scientific containers for mobility of compute. PLOS ONE 12(5): e0177459. DOI: 10.1371/journal.pone.0177459

