Grant agreement no. 675451

CompBioMed

Research and Innovation Action

H2020-EINFRA-2015-1

Topic: Centres of Excellence for Computing Applications

D5.5 Preparing data infrastructures for large-scale resources: report on the optimization activities

Work Package:	5	
Due date of deliverable:	Month 34	
Actual submission date:	July, 31 2019	
Start date of project:	October, 01 2016	Duration: 36 months

Lead beneficiary for this deliverable: *UEDIN*
Contributors: BSC, *SARA, USFD, UCL*

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.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

Table of Contents

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

List of Tables and Figures

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

1 Version log

Version	Date	Released by	Nature of Change
V0.1	20/11/2018	Charaka Palansuriya (UEDIN)	First Draft.
V0.2	06/03/2019	Charaka Palansuriya (UEDIN)	Highlighted sections where various partners could contribute.
V0.3	03/04/2019	Narges Zarrabi (SURFsara), Jazmin Aguado-Sierra (BSC), Charaka Palansuriya (UEDIN)	Merged additions from SURFSara and BSC.
V0.4	02/05/2019	Phil Tooley (USFD)	Added contributions from USFD (A description about the Diamond Light Source Pilot)
V0.5	12/06/2019	Steven Carlysle-Davies (UEDIN), Charaka Palansuriya (UEDIN)	Added optimisation work for using EPCC object storage, including the PolNet adaptation work.
V0.6	20/06/2019	Steven Carlysle-Davies (UEDIN), Charaka Palansuriya (UEDIN), Andrew Narracott (USFD)	Added testing done on EPCC object storage; added further details about use of GridFTP in Diamond Light Source Pilot. Added data replication scenarios using B2SAFE as suggested by Narges Zarrabi.
V0.7	26/06/2019	Narges Zarrabi (SURFsara), Steven Carlysle-Davies (UEDIN), Charaka Palansuriya (UEDIN)	Updates to “Testing Data replication with B2SAFE”. Updates to “Introduction”, “Conclusions” and the “Executive Summary”.
V1.0	28/06/2019	Terry Sloan (UEDIN), Charaka Palansuriya (UEDIN)	Addressed UEDIN QC comments and suggestions. Version sent for internal CompBioMed review.
V1.1	23/7/2019	Marco Verdicchio (SURFsara), Gavin Pringle (EPCC)	Revision to address the comments of internal reviewers.
V1.2	31/7/2019	Peter Coveney (UCL), Emily Lumley (UCL), Charaka Palansuriya (UEDIN)	Addressing final comments and suggestions from Peter Coveney and Emily Lumley

*“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No **675451**“*

2 Contributors

Name	Institution	Role
Charaka Palansuriya	UEDIN	Author
Steven Carlysle-Davies	UEDIN	Co-Author
Marco Verdicchio	SURFsara	Co-Author
Narges Zarrabi	SURFsara	Co-Author
Andrew Narracott	USFD	Co-Author
Phil Tooley	USFD	Co-Author
Jazmin Aguado-Sierra	BSC	Co-Author
Gavin Pringle	EPCC	Co-Author
David Wright	UCL	Reviewer
Okba Hamitou	BULL	Reviewer
Peter Coveney	UCL	Reviewer
Emily Lumley	UCL	Reviewer

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

3 Definition and Acronyms

Acronyms	Definitions
API	Application Programming Interface
AWS	Amazon Web Services
BoneDVC	Bone Digital Volume Correlation
BSC	Barcelona Supercomputing Centre
CFD	Computational Fluid Dynamics
CoE	Centre of Excellence
CPU	Central processing unit
DPM	Data Policy Manager
EOSC	European Open Science Cloud
ESRF	European Synchrotron Radiation Facility
EUDAT CDI	EUDAT Collaborative Data Infrastructure
FEM	Finite Element Method
GB	Gigabyte
GridFTP	Grid File Transfer Protocol
Hoff	HemeLB High Performance Offload Service
HPC	High Performance Computing
HPN-SSH	High Performance SSH
HTTP	Hypertext Transfer Protocol
iRODS	The Integration Role-Oriented Data System
MB	Megabyte
NFS	Network File System
OSG	Open Science Grid
PB	Petabyte
PID	Persistent Identifier
POSIX	Portable Operating System Interface
RDF	(UK) Research Data Facility
rsync	Remote file synchronisation tool
S3	(Amazon) Simple Storage Service
SCP	Secure Copy Protocol

SDK	Software Development Kit
SSH	Secure Shell
SSHFS	SSH File System
TCP	Transmission Control Protocol
URL	Uniform resource locator
WebDAV	Web Distributed Authoring and Versioning

“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No ***675451****“*

4 Executive summary

There is an ever-increasing demand in the biomedical community for storing more data as well as for the transfer, management and longer-term preservation of this data. These increasing data may be single large files (e.g., >1GB) or a large number of small files. A survey carried out earlier in the CompBioMed project revealed that data storage and handling will increase to hundreds of terabytes for some community members. The expectation is that the total size of the data to be stored and managed will exceed 2 petabytes within CompBioMed. This deliverable highlights the work carried out to satisfy these requirements by the provision of object storage with a capacity of over 2.5 petabytes (PB). Testing done on this object storage shows that it performs well for transferring and managing both large numbers of small files (< 4Mb) as well as for single large files (e.g., 4GB files). The PolNet application was adapted to use this object storage and shows one of the many ways that biomedical applications could use the large reliable storage capacities provided by object storage systems.

Generally, large data sets need to be moved closer to High Performance Computing (HPC) services prior to performing computational work. Once the computational work is done, the resulting data is then moved somewhere else or kept closer to the HPC services for post-processing work. This data may be stored for the short to long term for various reasons (e.g., comparison with future computational work, to refer to them from publications or to make it findable). Suitable data services as well as data transfer tools and APIs for performing these types of tasks are briefly described in this deliverable. In particular, large data transfer activities and tests carried out using GridFTP and Amazon S3 API are described. The Diamond Light Source Pilot illustrates how GridFTP could be used to optimise bandwidth usage during large data transfers. GridFTP benchmarking of transfer rates between the Diamond and RDF endpoints was undertaken with datasets of hundreds of gigabytes. This provided rates in the region of 400MB/s for data transfers with appropriate use of parallel data channels. Third party tools that use Amazon S3 API (i.e., COSBench and Benchio) were used to perform data transfers to and from the EPCC object storage. As shown in the testing performed between user machines and the EPCC object storage, it too can achieve rates of many hundreds of MB/s using parallel data channels. For effective bandwidth usage it is necessary to transfer data in parallel using multiple parallel workers (i.e., using multiple parallel data channels). In many cases, the users themselves would not have to choose whether to use parallel workers or not – many S3 clients (although not all) will automatically use parallel workers, and so the user only has to make sure they are using a suitably full-featured Amazon S3 client.

CompBioMed users may benefit from using the EUDAT Collaborative Data Infrastructure (CDI) for data transfer, staging, long term storage and automated replication and backing up of data. EUDAT guarantees long-term persistence of data and allows data to be shared worldwide. One of the advantages of using EUDAT B2SAFE service is how the data could be replicated between geographically distributed data storages closer to HPC resources. The optimisation work carried out using the B2SAFE service to replicate data this way is described here.

This deliverable also briefly describes a list of suitable data services available to CompBioMed users from EUDAT CDI, SURFSara and EPCC. Such users should consider these data services when creating a data management plan for their particular research work.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

5 Introduction

One of the clear trends in the biomedical community is its ever-increasing demand for storing more data as well as the transfer, management and longer-term preservation of this data. In fact, the CompBioMed deliverable D5.2 (Report on computing and data needs of the biomedical community) [1] highlights the following clear requirements with respect to the increasing data volumes:

- "The size of non-simulated data that is typical of that used in computational biomedicine may vary from a few Megabytes (e.g. molecular structures files and/or molecular dynamics input files), to a few hundred Gigabytes (e.g. medical images processed in cardiovascular and neuro- musculoskeletal modelling) and the dimensions of a single file are directly proportional to the size of the investigated problem."
- "Many users within the CompBioMed community, require the storage of a large number of input files for each simulation, resulting in significant disk space requirements even in the case of small input files."
- For simulated data, typical output ranges from few Gigabytes to tens of terabytes.

A survey carried out earlier in the CompBioMed project revealed that data storage and handling will increase to hundreds of terabytes in the near future for some community members. As mentioned in the CompBioMed data management plan (i.e., D1.3) [2], it is anticipated that the total size of the data to be stored and managed will exceed 2 petabytes. This deliverable will highlight the work carried out to satisfy these requirements.

Frequently, large data sets need to be moved closer to High Performance Computing (HPC) services prior to performing computational work. Once the computational work is done, the resulting data is then moved to somewhere else or kept closer to the HPC services for post-processing work. This data may be stored in the short to long term for various reasons (e.g., comparison with future computational work, to refer to them from publications or to make it findable). Suitable data transfer tools and APIs for performing these types of tasks are briefly described in this deliverable. In particular, large data transfer activities carried out using GridFTP and Amazon S3 API are described.

As highlighted in D1.3, CompBioMed users may benefit from using the EUDAT Collaborative Data Infrastructure (CDI) [3] for data transfer, staging, long term storage and automated replication and back-up of data. EUDAT guarantees long-term persistence of data and allows data to be shared worldwide. It creates persistent identifiers for data, and together with the metadata, makes hosted data findable to others using the B2FIND EUDAT service. Another clear advantage is how the data could be replicated between geographically distributed data storages closer to HPC resources. The optimisation work carried out using the B2SAFE EUDAT service to replicate data is described in here.

6 Data Services

This section briefly describes a set of data services offered to the CompBioMed community. In particular, it lists available data storage and transfer solutions and, where appropriate, points out any particular limitations to commercial users.

It is expected that usage of many CompBioMed applications will either require utilisation of and/or generation of large amounts of data (e.g., medical images). In order to utilise a large

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

amount of data in an effective way on an HPC platform, the data need to be either on the HPC platform itself or very close to it. The following sub-sections highlight such data services available to the CompBioMed community.

6.1 EUDAT data services

EUDAT provides various data storage services. Some of the services are suitable for sharing data whilst others are suitable for long term storage or archiving. To use EUDAT data services one needs to register and have B2ACCESS user credentials. EUDAT data services are described at https://www.eudat.eu/services.

It should be noted that Barcelona Supercomputing Centre (BSC) contributes resources to EUDAT data services and this is one of the ways to use BSC data infrastructure.

6.1.1 B2SAFE

The B2SAFE is a service for defining data management policies on research data that is distributed across different geographical and administrative domains. The B2SAFE service offers functionality to replicate data across different data centres in a safe and efficient way. The service makes use of persistent identifiers (PIDs) to maintain all information required to find and query information about the replica locations. Federation between remote sites is based on iRODS (http://www.irods.org). The iRODS middleware is also used to replicate datasets from a source data (or community) centre to a destination data centre.

6.1.2 B2DROP

This service provides up to 20 GB of storage space to share data with others. This could also be used to store input/output data that is required/generated by HPC applications. The services allow scientists to keep their data synchronised and up to date.

The B2DROP service is open to all researchers, scientists, communities alike to synchronise and exchange data. Daily backups of all files in B2DROP are taken and kept on tape. B2DROP is hosted at the Jülich Supercomputing Centre, which guarantees that users' research data stay in Europe and are accessible for at least 6 months. EUDAT makes no claim over the data held in B2DROP and uploaders remain entirely responsible for the data they upload and share.

6.1.3 B2SHARE

The B2SHARE service is an EUDAT data repository service for storing, preserving and publishing research data with metadata. B2SHARE offers storage and sharing of small-scale stable research data. The data is assigned a permanent identifier which can be traced to a data owner. This is a professionally managed and monitored data storage system where scientists do not have to worry about long term storage of their data.

The data being published in B2SHARE should be static and in a final state. The data owner defines the access policies to the data. When publishing data, you can choose whether the files are openly accessible or not. When restricted access is chosen, the uploaded file will only be accessible by the data owner and community administrators, while the metadata is always publicly available.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

The B2SHARE service is available via https://b2share.eudat.eu and is hosted in Finland. This service is open to all researchers and scientists in Europe and can be accessed with B2ACCESS user credentials.

6.1.4 B2STAGE

B2STAGE offers data staging services for the transfer of large data sets between remote HPC centres and the EUDAT data storage services and HPC facilities. The B2STAGE service uses GridFTP to transfer the data. On the client side, any client supporting the GridFTP protocol can be used, such as globus-url-copy, Globus Online, UberFTP. EUDAT also provides a script to facilitate the integration of B2STAGE within existing community solutions, such as web portals, workflow engines, etc. The Data Staging Script, as well as providing common data staging functionalities, also allows the retrieval of PIDs assigned to ingested data.

The B2STAGE service can be used by all EUDAT registered users (i.e., researchers and interested communities). The users should negotiate access to remote HPC service in parallel.

6.2 SURFsara data services

SURFSara offers a variety of data services, including data storage. A list of data storage and other services offered by SURFSara is available from https://www.surf.nl/en/about-surf/subsidiaries/surfsara/. A few relevant data storage services are listed below.

6.2.1 SURFdrive

SURFdrive is an online storage and data sharing service based on OwnCloud. SURFdrive allocates 250GB storage capacity per user. Users can access their files from anywhere, using any device at any time via secure encrypted connections. The service offers offline synchronisation of files and secure file sharing.

This is a premium paid service and is accessible to all members of universities and research institutes in the Netherlands. With SURFdrive, sharing and exchange of data with international collaborators is possible through a guest account functionality. Institutions can choose between two different plans:

- The first package costs EUR 172 per month. This package includes 50 accounts and there is a EUR 3,35 per month surcharge per each additional account.
- The other package costs EUR 2153 per month and it includes 750 accounts. For every extra account, users pay EUR 1,50 per year.

6.2.2 Data Archive

This is a service for a long-term data storage. A user can securely store data up to petabyte-scale in size. This data storage provides quick access to HPC platforms at SURFSara via NFS mounts. The service supports multiple protocols for transferring data including SCP, rsync and GridFTP.

The data stored on tape is backed up in two physical locations. Users who have HPC access at SURFsara, have automatic access to the Data Archive. The service can also be accessed via external contracts.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

6.2.3 Object Storage

The SURFsara object storage service allows general purpose online storage of large quantities of data. The data is stored as objects in a flat structure of containers (this is in contrast to a traditional file system storage which organises data in a hierarchy of folders and files). The Object Storage can scale up far beyond the capabilities of file-based storage systems. Access is via the OpenStack Swift and Amazon S3 protocols. This means that existing tools that support Amazon S3 API can be used to access the service.

6.2.4 Research Drive

This is a service for storing and sharing large research data. Research Drive is based on OwnCloud and the storage backend is the SURFsara object storage [4]. The service provides data synchronisation across different devices. Research Drive does not make regular backups of the data. The service uses the WebDAV protocol and is accessible from all systems with an HTTP client installed.

Accessing the service is possible via two routes. If the compute infrastructure at SURFsara is being used (such as HPC cloud, Lisa, Cartesius), the user can apply for the service via an e-infra contract, by filling out an e-infra request form. The service can also be accessed independently for data storage and sharing. In this case the user needs to make a separate contract and there will be fees involved. To buy the service, you would need to send an email to the SURFsara helpdesk (helpdesk@surfsara.nl).

6.3 EPCC data services

6.3.1 UK Research Data Facility (UK-RDF)

UK RDF provides a high capacity robust file storage system (http://www.archer.ac.uk/documentation/rdf-guide/). It is a long term persistent storage system that will last beyond any national HPC service like ARCHER (UK national supercomputing service).

The UK-RDF is a paid data storage service open for both academic and commercial customers.

6.3.2 EPCC Object Storage

Object storage is being commissioned at EPCC for large scale data storage. Due to its physical proximity and the fast network connections to the Cirrus HPC platform, this is likely to become a suitable storage service for use with Cirrus. Access is via Amazon Simple Storage Service (S3) protocol. That is, any existing tool that supports Amazon S3 API could be used to access the service once available.

The EPCC Object Storage will be a paid data storage service open for both academic and commercial customers.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

7 Data Transfer Tools and APIs

7.1 SCP

The Secure Copy Protocol (SCP) is almost universally supported on Unix-like operating systems. This protocol is used to securely transfer a file (or a directory containing many files) from a local machine to a remote machine. The `scp` command line tool implements this protocol and can be used to transfer files between the storage services mentioned above and HPC platforms like ARCHER, Cirrus and Cartesius.

7.2 rsync

The remote file synchronisation tool, `rsync`, is also supported by virtually all Unix-like operating systems. Like the `scp` tool, `rsync` can be used to transfer a file (or a directory containing many files) from a local machine to a remote machine. When used for the first time `rsync` behaves like `scp`; however, if the files are updated or new files are added then any subsequent use of `rsync` will only copy the updates or just the newly added files. Therefore, `rsync` is a very efficient and versatile file transfer tool.

7.3 GridFTP

GridFTP is a high performance data transfer protocol that is based on the FTP protocol. It is optimized to work with high bandwidth wide area networks. The most widely used GridFTP tool is the one provided by the Globus Toolkit. It uses the available bandwidth much more effectively than many other data transfer tools by using multiple simultaneous TCP streams. The support for GridFTP is provided by the Open Science Grid (OSG), http://opensciencegrid.org/technology/policy/globus-toolkit/

GridFTP should be considered if very large files are to be transferred to and from HPC platforms. Many HPC platforms have GridFTP deployed as an end point. For example, UK-RDF has a GridFTP endpoint. Since there is a fast network connection between UK-RDF and ARCHER, the use of GridFTP provides the fastest way to transfer large files to and from ARCHER.

7.4 Amazon S3 REST API

Data transfers to and from object storages such as the EPCC object storage and SURFSara object storage will be supported using Amazon Simple Storage Service (S3) API. This means that an existing Amazon S3 client (e.g., AWS CLI or s3cmd) could be used for efficient fast data transfers or one could be developed or integrated into your existing data management tools. These object storages may also support the OpenStack's Swift Object Storage API.

7.4.1 Cyberduck

Cyberduck (https://cyberduck.io/) is a graphical data storage browser that can be used to access object storages that support Amazon S3 API. Therefore, this could be used to browse and access data stored in SURFsara and EPCC object storages.

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No ***675451***"

8 Optimisation Work

This section describes the optimisation work carried out using the selected CompBioMed applications.

8.1 Selected Applications

8.1.1 Alya

Alya, developed by the team of Mariano Vazquez and Guillaume Houzeaux at the Barcelona Supercomputing Centre, performs multi-scale, multi-physics biomechanical simulations. The specialties of Alya regarding biomedical sciences include cardiac electro-mechanic-flow simulations, from ion-channel kinetics up to organ level; and simulations of the respiratory system, particularly focusing on particle deposition. The simulations involve the solution of mathematical models at different scales using a FEM-based approach. Alya has been specifically optimised for the efficient use of supercomputing resources and does not employ external libraries. Alya is available for use by research scientists on MareNostrum, ARCHER, and Cartesius; for clinical and industrial users, BSC recommends users access it as a service, due to the complexity involved with setting up simulations. To this purpose BSC has launched a spin-off (ELEM Biotech [5]) that will provide commercial software-as-a-service to medical device, pharmaceutical and biomedical industries using Alya.

8.1.2 HemeLB

HemeLB, developed by the team of Prof Peter Coveney at University College London (UK), is a software pipeline that simulates the blood flow through a stent (or other flow diverting device) inserted in a patient's brain. The aim is to discover how different stent designs (surface patterns) affect the stress the blood applies to the blood vessel, in particular in the region of the aneurysm being treated. The pipeline also allows the motion of magnetically steered particles, for example coated with drugs, to be simulated and estimates made as to where they might statistically end up. The HemeLB setup tool voxelises the geometry at the given resolution, and HemeLB (lattice-Boltzmann CFD solver) then simulates the fluid flow within that geometry, using the given velocity-time profiles for each inlet. Once complete, the simulation output is analysed using the hemeXtract utility, which can produce images of cross-sectional flow, or 3D shots of wall shear stress distribution in the geometry using ParaView visualisation software. HemeLB is installed, optimised, and available for use to any user with a valid account and CPU-time on ARCHER, Cartesius, SuperMUC, Prometheus and Blue Waters. The UCL team also provides consulting to biomedical companies and clinical users.

8.1.3 PolNet

PolNet is a software tool for the computer simulation of blood flow in realistic microvascular networks imaged with a wide variety of microscopy and clinical imaging techniques. To date, PolNet has contributed to: a) uncovering the relationship between blood flow and blood vessel biology and its importance for correct vascularisation of tissues, and b) developing ways of predicting retinal vascular damage in diabetic retinopathy patients. PolNet facilitates the adoption of cutting-edge computer simulation technology by non-experts in the Biosciences.

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No ***675451****"*

8.2 Optimisation work with EUDAT services

The data-intensive workflows and distributed international partners involved in the CompBioMed project necessitate the use of proper data management solutions for handling large data over distributed sites. EOSC (European Open Science Cloud) and EUDAT offer data services and expertise that can be used for this purpose.

8.2.1 Data replication use case with B2SAFE

In collaboration with the EOSC-hub project [6], we have worked on a use case for data replication using the EUDAT B2SAFE service. This use case addresses the need for safe data replication and large data transfer, which is an important data requirement within this international community. We started with identifying the requirements. The solution encompasses the definition of a data pipeline and of the related data policy as described below. Once the replication service is setup and configured, we expect to replicate terabytes of data between the HPC centres and so facilitate large data exchange and access to valuable data for researchers in this community.

The HPC centres involved are BSC (Barcelona Supercomputing Centre), SURFsara (Netherlands) and EPCC (UK). The resources include allocation of at least 22 TB storage at each of the HPC centres.

Requirements:

- Data to be replicated is 3D finite element mesh (file format can be .vtk, .txt).
- The maximum size per file is 1.2 TB.
- The total data to be replicated is not more than 24 TB.
- Two copies of replicas are desired, one on the compute facilities to run simulations and one on tape.
- The data owner assesses the replicas.
- Data will be downloaded by researchers
- Full access control to the data (i.e. read/write/exec access)
- Data needs to be findable potentially after publication and/or after the 3-year quarantine

Data Pipeline:

The data pipeline includes the following major steps:

Step 1: Data creation and transfer: The raw data is collected at ESRF (European Synchrotron Radiation Facility) in France. The data is being stored locally on tapes. Currently, a copy of the data had to be transferred to BSC on hard disk drives.

Step 2: Data pre-processing: In BSC, researchers pre-process the data which includes manual and automated steps for image stitching, segmentation and meshing

Step 3: Data Replication: The pre-processed data needs to be replicated from BSC to SURFsara and EPCC. The replicated data will then be used to run simulations with the Alya software which is installed on the supercomputers in these sites (e.g., Cartesius in SURFsara).

Replication policy:

*“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No **675451**“*

The replication policy for the CompBioMed use case was defined in the EUDAT Data Policy Manager (DPM) tool [7]. The generic replication policy schema of B2SAFE covered the replication requirements of the community, including having two replicas of data on both disk and tape.

Figure 1 below shows the data workflow, services and centres that are involved.

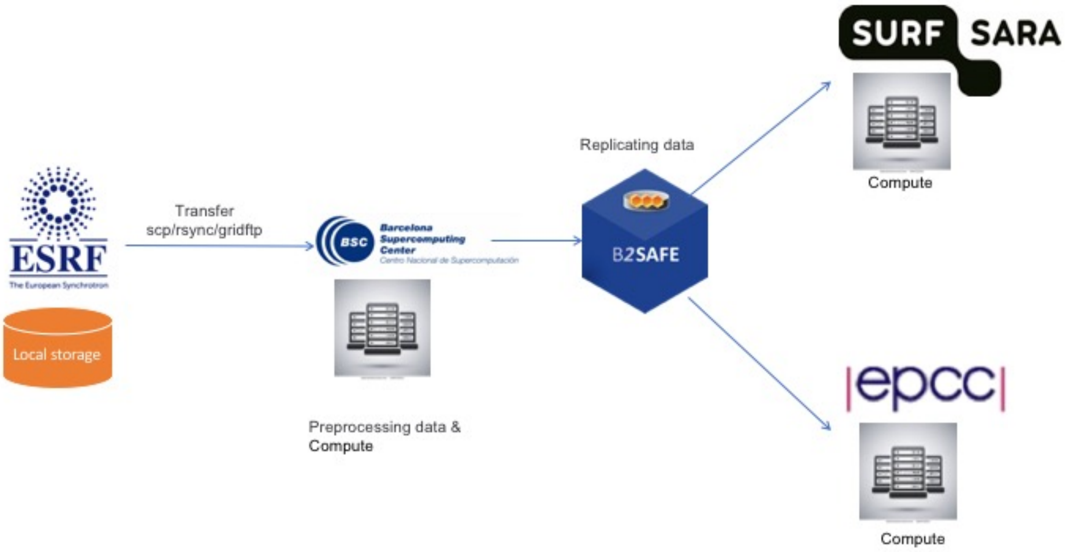

Figure 1: The data workflow and services involved in the use case.

8.2.2 Testing Data Replication with B2SAFE

For the data replication pipeline with B2SAFE, two specific scenarios have been defined. The first scenario involves the replication of data set from one HPC centre to two different end-points, triggered by user interaction. The second one, involves replication of the data, from one origin, to a single end point which then automatically replicate the data to a second HPC centre. The two scenarios will be used to investigate different topologies of the data transfer workflow, which can be combined to satisfy more complex user needs.

8.2.2.1 First Data Replication scenario

In this scenario, data is replicated from BSC to EPCC and from BSC to SURFsara. First, BSC generates random data and triggers the replication to the EPCC and SURFsara destination nodes as shown in Figure 2.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

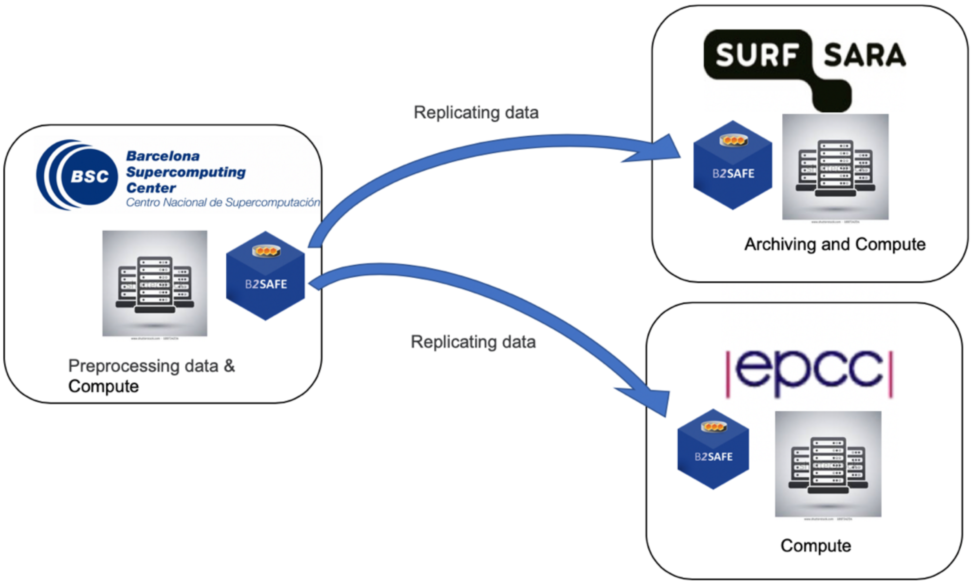
Barcelona Supercomputing Center
Centro Nacional de Supercomputación
B2SAFE
Preprocessing data & Compute
Replicating data
SURF SARA
B2SAFE
Archiving and Compute
Replicating data
epcc
B2SAFE
Compute

Figure 2: B2SAFE data replication - First Scenario

8.2.2.2 Second Data Replication scenario

In this scenario, data is replicated from BSC to EPCC and from EPCC to SURFsara. First, the source node at BSC generates random data and triggers the replication to EPCC. Then the middle node at EPCC triggers the replication to SURFsara. In this case, SURFsara is the end node which is also an archiving node. Figure 3 shows this scenario. The idea of this scenario is to have the overall replication done in an automatic way. Implementing this scenario requires development of iRODS and B2SAFE rules for automatic triggering of replication. This work is still in progress.

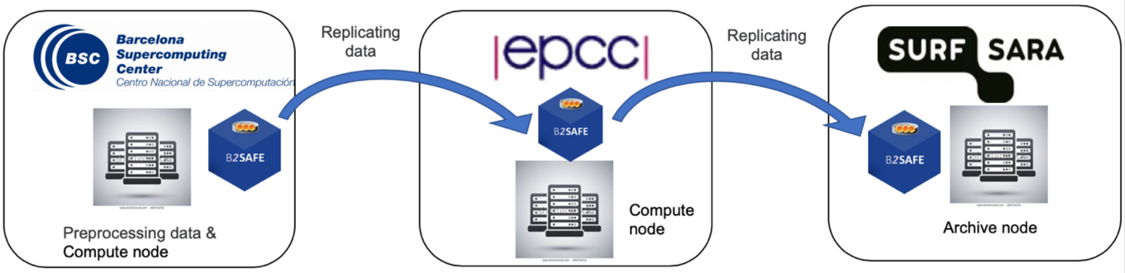
Barcelona Supercomputing Center
Centro Nacional de Supercomputación
B2SAFE
Preprocessing data & Compute node
Replicating data
epcc
B2SAFE
Compute node
Replicating data
SURF SARA
B2SAFE
Archive node

Figure 3: B2SAFE data replication - Second Scenario

8.2.2.3 Performance measures

The performance of data replication with B2SAFE is influenced by the data organization and file sizes of the datasets ingested or restored from the system. As a performance measure we calculate the Time of Replication which includes creation of checksums, creation of PIDs and the actual transfer of the data. We performed several test transfers for randomly generated data with a volume of 30GB between two HPC centres. Preliminary results show a Time of Replication of about 6m30s for transfers between SURFsara and BSC (transfer rate ~240 MB/s), 10m50s for transfers between SURFsara and EPCC (transfer rate ~110 MB/s) and, at the time of writing this deliverable, we are awaiting for the final timings and transfer rates between BSC and EPCC . The different times of replication cannot be simply explained by the difference in transfer rates, but

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

it is associated to different performances for the creation of checksums. In these tests, in fact, more than 50% of the total transfer time was used for the creation of the checksums. At the time of writing, this work is still in progress and we are planning to investigate further ways to improve performances as well as to test transfers of larger data sets.

8.3 Optimisation work with Object Storages

8.3.1 What are Object Storage systems?

Object Storage systems do not contain any concept of files or directories, instead storing data as objects, where each object has some form of global identifier. Objects can be written, read or deleted, but importantly cannot be updated. Objects are therefore immutable - the only way to modify an object is to replace it (although some systems allow appending).

Most object storage systems then add an additional interface in front of the object-storage that support additional features. For example, the Lustre file system, commonly used within many supercomputers (including Cirrus), offers a traditional POSIX-compliant file system on top of underlying object storage servers. While this makes it possible for end-users to treat object storage systems in the same way as other file systems, these interfaces can themselves be complex, and using them does not allow users to take advantage of the simplicity of object storage. Many other systems offer other APIs instead, usually HTTP-based, which are closer to the way the underlying storage works. These APIs may be specific to the object storage system, but there are a few commonly used APIs such as OpenStack Swift or AWS S3. This document will look at object storage systems with such APIs.

For most systems, objects are placed within buckets which can be used for namespacing and access control requirements. Some systems may allow a **"/"** character to be used as part of an object identifier, and client software may interpret this to group objects with the same prefix into a folder, allowing the system to appear like a more traditional file system. It is important to note that this is still part of the identifier, however, and no actual file hierarchy exists.

8.3.2 What does an Object Storage system allow users to do that could not be done with other storage technologies?

The restrictions of object storage technology make it much simpler than other types of storage. This leads to reduced costs. The principal benefit of object storage technology for the biomedical HPC community therefore is scalability and reduced cost, allowing a much greater volume of storage to be offered to projects.

To take advantage of this, researchers will have to use additional APIs to access the storage. While this could be considered as a disadvantage, these HTTP interfaces are also a major benefit for some use cases. While it is possible for traditional file systems to be accessed remotely, this often involves the use of tools that may be complex, and not always suitable for all applications, such as NFS or SSHFS. If objects are available via a commonly used API, e.g. AWS S3 or Swift, this opens the possibility for the same objects to be used in simulation runs on multiple HPC systems, on local systems and even on public Cloud infrastructures like AWS.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

Additionally, for some workflows, object storage may be faster than file storage, particularly those that match the intended pattern of writing whole objects once, then reading them multiple times. Some object stores claim to offer faster access by storing multiple replicas across different servers. Access requests can then be routed to the server with the lowest latency based on the origin of the request. Replicas can also be migrated between servers to minimise access time.

Object storage systems also offer a different approach to permissions than traditional file systems, which may be more suitable for some applications. Many HPC systems are configured with a simple group-based system, where all members of a group are allowed to access data and to perform computations. An object storage system could more easily allow this to be separated, so that some users may be allowed to upload or retrieve data, but not necessarily submit jobs. It also allows for objects to be shared publicly, or to generate temporary URLs which can be shared allowing access for a limited length of time.

There are some additional benefits that may be suitable for some use cases. Some object stores allow easy specification of notifications upon events, e.g. triggering some action when an object is uploaded to a particular bucket. Some also allow users to associate arbitrary metadata with an object, which can be used for multiple purposes e.g. retrieved independently of the object or used to help search objects.

None of these things are impossible with traditional file systems, but most object storage systems are designed with them in mind, so it is significantly easier.

8.3.3 How can existing codes be adapted to work with object stores?

It is important to note that object stores are designed for particular use cases and so will not necessarily be appropriate for other uses. In particular, they are not designed for updating objects, but only for reading and writing (including replacing) objects. Codes that do a lot of seeking to arbitrary positions in a lot of files will not necessarily adapt well. That said, it is anticipated that many codes do not have these requirements.

Researchers have a number of options for working with object stores:

- Using codes that can read from or write to an object store directly
- Adapting their code to do so
- Using tools to copy objects to the local disk before running the codes, then uploading the results afterwards
- Using tools to stream objects from the store into the codes, then streaming the resulting output back to the store

8.3.4 Using EPCC Object Storage

EPCC deployed an object storage during the CompBioMed project, to provide greater data storage capacity near to the Cirrus HPC platform at EPCC. Currently, over 2.5 Petabytes of storage is available. The object storage technology provides highly scalable and highly fault tolerant storage capabilities. The access is through the de facto standard AWS S3 REST API. Due to the close proximity and fast network access to the Cirrus HPC platform at EPCC, this will be a suitable data service to be used if access to large data storage is required for pre or post HPC

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

workloads on Cirrus. The service is accessible from cirrus-s3.epcc.ed.ac.uk once an account is created for a user.

8.3.5 Adapting PolNet to use EPCC Object Storage

PolNet (see description in section 8.1.3) as a whole is a workflow which runs inside a Docker container on the researcher's machine and includes several stages. The system takes microscopic and clinical images, generates a model of the blood flow, allows the researcher to specify locations of objects, simulates the blood flow using **HemeLB** (see description in section 8.1.2), and interprets the output to produce charts.

As part of CompBioMed, EPCC have developed a service called the HemeLB High Performance Offload service (Hoff) which offloads simulation jobs to an HPC system, hence significantly reducing the time they take. The remainder of this description will illustrate the workflow using Cirrus as the HPC system, but Hoff can be configured with multiple HPC systems, between which users can choose.

To use the Hoff, the client will upload their input files to the server, which copies them across to the file system on Cirrus. The Hoff then schedules a simulation on Cirrus. When the simulation runs, it will read the input files from the file system, then write the output to the file system. Once the simulation is complete, the Hoff will retrieve the files from the Cirrus file system, and store them in its own storage, and notify the client the job has completed. The client then has to retrieve the files from the web server, and then delete the job, which will allow the storage to be reclaimed.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

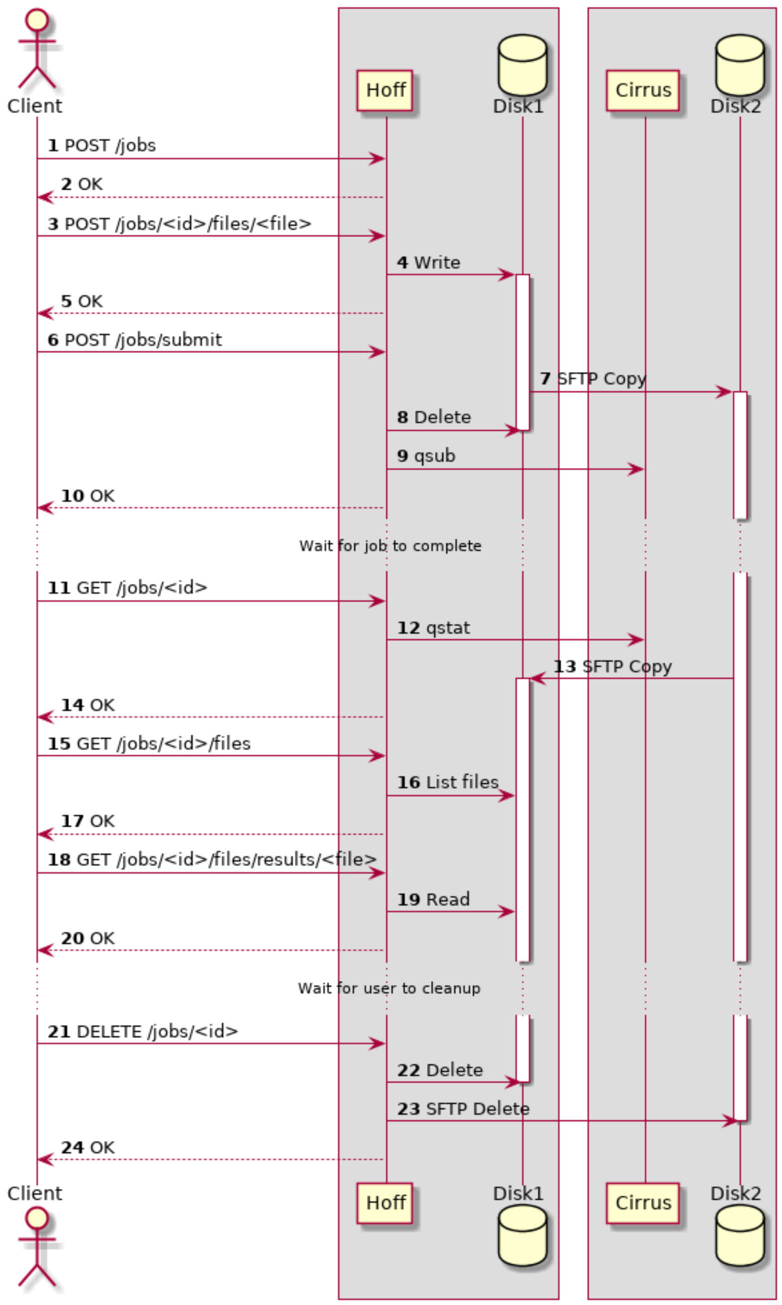

Figure 4: Sequence diagram showing Hoff using the file system in Hoff server and Cirrus

As shown in Figure 4, the input and output files for the simulation remain on the disk of the Hoff server until the user deletes them. While the input files are relatively small, the output files are significantly larger. If the client does not delete the files, they will either remain indefinitely (taking up storage space) or will have to be cleaned up by an automated process (which may result in users losing access to files and having to re-run simulations). The size of the local disk acts as a limit on the number of users who can use the system at one time (separate to any resource constraints on the HPC system), and as such the Hoff limits the number of ongoing jobs fairly strictly.

As a proof-of-concept prototype, we have adapted the Hoff to work with the EPCC object storage system installed on Cirrus. As shown in Figure 5, after retrieving the simulation output files, a

*“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No **675451**“*

bucket is created within the object storage, the files are copied inside it, and the local copy deleted. When a user requests a file, the file is downloaded from the bucket (temporarily storing it locally), and then sent to the user. Further technical details and source code for this proof-of-concept prototype is available under the CompBioMed GitHub organisation, see https://github.com/compbiomedeu/hemelb-hoff.

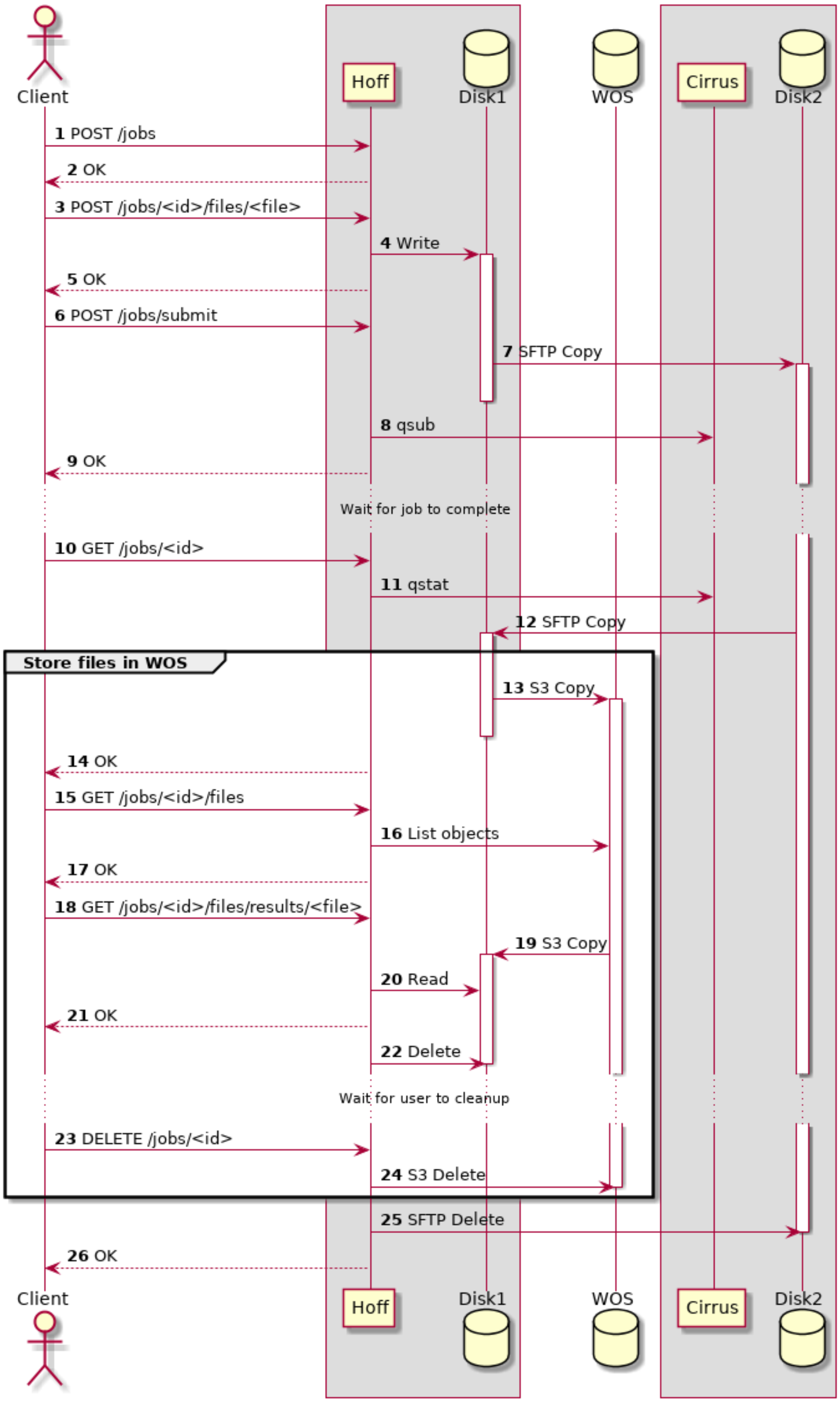

Figure 5: Sequence diagram showing Hoff using the EPCC object storage (i.e., WOS) and Cirrus

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

In some ways, this can be considered as simply shifting the problem: instead of the files taking up space on the Hoff server until the user deletes them, they take up space on the object storage. Additionally, there are now additional network transfers involved. However, the resources available on the object storage are significantly larger than those on the hardware available for the Hoff server (e.g., many petabytes of storage space).

As a further advancement, as shown in Figure 6, it is also possible for the user to retrieve the files directly from the object storage system, bypassing the need to temporarily download them to the Hoff. This is achieved through the use of a pre-signed URL. These allow objects in an object storage to be shared with users who would not ordinarily be able to see them, for a limited time.

“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No ***675451****“*

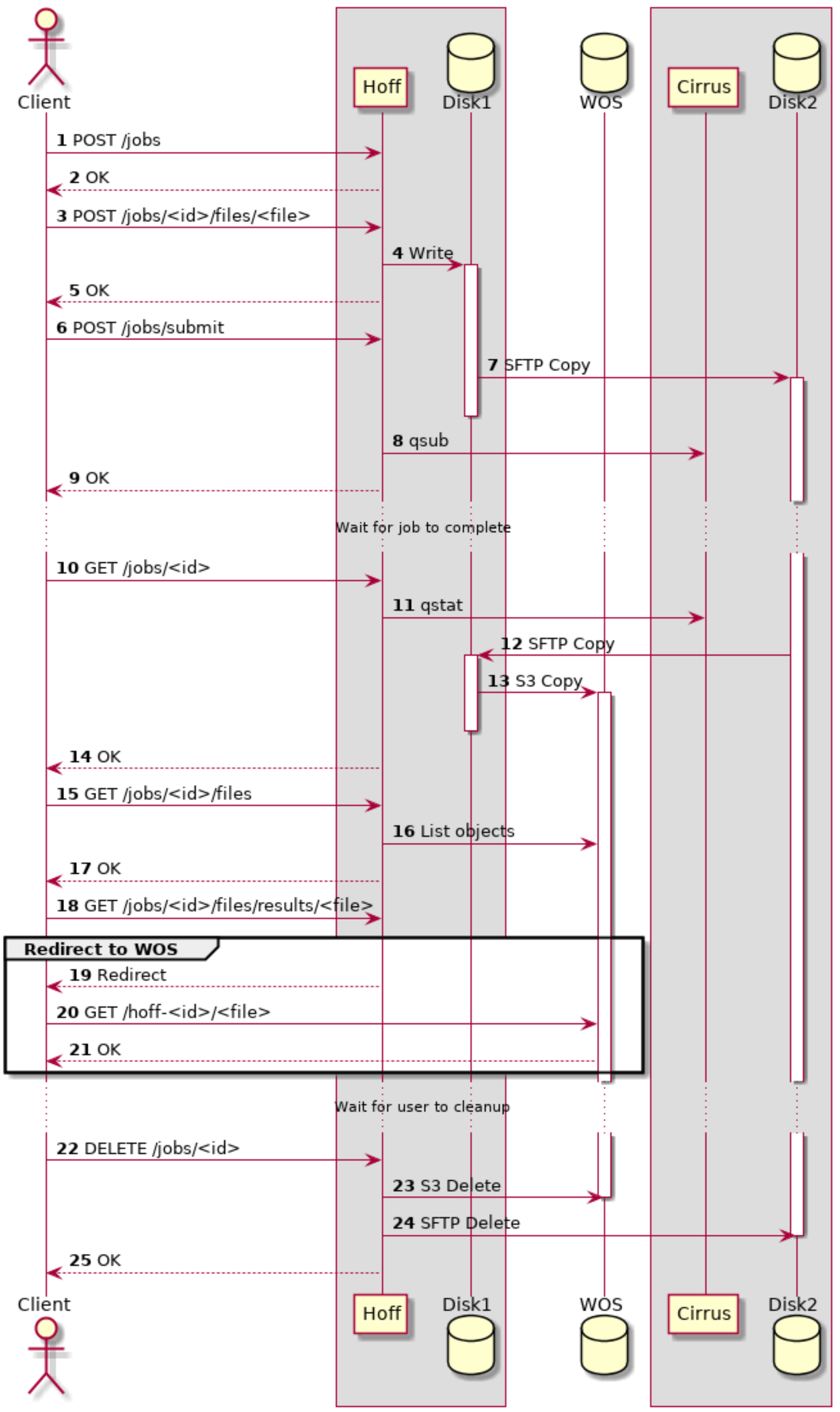

Figure 6: Sequence diagram showing Hoff user directly retrieving file from the EPCC object storage

Other further improvements are also possible as follows:

- Files could be removed from the HPC system after being copied to the object store, rather than when the user deletes (i.e. execute step 24 after step 13. See Figure 6)

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

- Using pre-signed URLs, scripts could be executed on the HPC system that copy objects from the HPC file system to the object storage, without the need to copy them to the Hoff server first (e.g. altering steps 12 and 13 in the last diagram)
- Pre-signed URLs could be used by the client to supply the Hoff with the location to stored files. This would then mean that the storage on the object storage was under control of the user, not the Hoff. This would eliminate the need for the deletion steps (22+) entirely.
- The last two steps could also be achieved through passing credentials to the appropriate system, but that has a significantly higher security risk.
- It would also be possible for the users to upload their input files to the object storage, from where they could be retrieved onto Cirrus, although since the input files are small this does not have a significant benefit.

8.3.6 Testing of EPCC Object Storage

Once the EPCC object storage was provisioned, through CompBioMed WP5, it was tested comprehensively using third-party object storage testing tools. These tests focused on accuracy, reliability, performance as well as suitability for typical CompBioMed usage. The following tools were used:

- COSBench, https://github.com/intel-cloud/cosbench
- Benchio, https://github.com/giacomoguiulfo/benchio

8.3.6.1 COSBench

COSBench is a tool for benchmarking object storage systems. It claims support for a variety of object storage implementations, including AWS S3. It has a distributed architecture, designed to enable test runs on multiple machines.

It is flexible and supports running multiple workers in parallel doing a mix of operations, e.g.:

- 8 workers, each reading and writing to the same set of buckets, with a 4 to 1 ratio of read-write operations.
- 2 workers reading small objects, 2 workers writing small objects, 2 workers reading large objects, 2 workers writing large objects.

By setting the `hashCheck=true` parameter on a write operation, COSBench will add a checksum of the data to the object before writing in, and by setting the same parameter on a read, it will verify any checksums it finds in downloaded objects. This is used to check that the data stored in the EPCC object storage is the same as expected (i.e., to check the accuracy of the object stored within buckets).

8.3.6.2 Benchio

Benchio is another tool for benchmarking AWS S3 systems. It is much simpler, but more limited than COSBench. It only supports S3 interfaces, and all tests follow the same process: write a number of objects of a particular size to a single bucket, then read back those objects. The number of objects, size of objects and number of parallel workers is configurable but is the same throughout the test. It does not carry out any verification that the contents of the object that is received are the same as that sent, although it does check the size.

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"

Benchio has been designed for testing large numbers of small objects, and not for testing large objects, as might be used by typical CompBioMed users. As such, we forked Benchio to make alterations for this use case:

- Allow for the execution of tests of objects larger than the memory of the local machine. Previously, Benchio would create a random data set of the specified size in memory at the start, then upload it repeatedly. Our change allowed it to read the data multiple times, allowing tests of theoretically unlimited size.
- Use AWS S3 multi-part uploads, which allows large objects to be uploaded as multiple HTTP requests.

8.3.6.3 Are you able to upload and download lots of smaller objects in parallel?

One of the key requirements identified by CompBioMed D5.2 (Report on computing and data needs of the biomedical community) is that the CompBioMed community users require the storage of a larger number of input files for each simulation, resulting in significant storage requirements even in the case of small input files. Typically, a user would like to transfer such a large number of files (i.e., objects in the case of object storage) as fast as possible. This would mean transferring objects in parallel using multiple worker tasks.

A series of tests were carried out using COSBench from a personal laptop. Each ran for 15 minutes, doing a 50-50 mix of reads and writes of objects between 1-4MB to a particular server, with a varying number of workers.

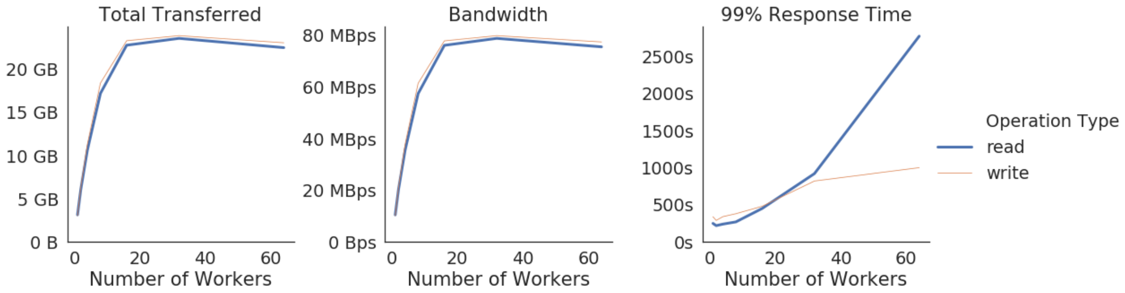

Figure 7: Small object transfers: as the number of workers increases, performance improves, plateaus, then decreases.

As shown in Figure 7, the total amount transferred and the bandwidth used increased when comparing a single worker to a small number of parallel workers. As the number of workers increases further however, there appear to be two bottlenecks preventing the amount transferred from increasing further - first the network connection, then the number of cores on the test machine.

In many cases, the researchers themselves would not have to choose whether to use parallel workers or not – many S3 clients (although not all) will automatically use parallel workers, and so the researcher only has to make sure they are using a suitably full-featured client. When using the S3 API directly, or through the various SDKs, they will need to be aware of the issue and write their code appropriately.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*

8.3.6.4 Are you able to upload and download large objects in parallel?

A further series of tests were carried out using Benchio from a personal laptop. Each wrote 16 4GB objects to a particular server, followed by reading back the same objects, with a varying number of workers.

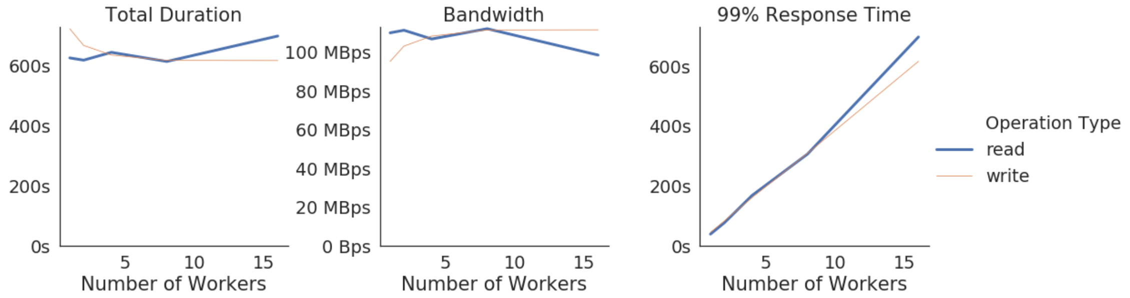

Figure 8: Large object transfers: as number of workers increases, performance is initially unchanged then decreases

As shown in Figure 8, with larger objects, there is little advantage to using more workers - one worker could use all of the available bandwidth. It should be noted though that the performance of the systems does not seem to suffer from handling multiple uploads simultaneously.

8.4 Additional optimisation work

8.4.1 The Diamond Light Source Pilot

As described in detail in CompBioMed deliverable D6.3 (Report on Workflow system provision), large scale data transfer between the Diamond Light Source and the EPCC Archer HPC facility is needed to perform the BoneDVC workflow. This uses the GridFTP data transfer infrastructure and the Globus framework for authentication. Data transfer is requested by the user from a client machine and all authentication to the Diamond and EPCC systems is performed through the Globus API.

Here we briefly review the details reported in deliverable D6.3 in the context of the wider development of data infrastructures within CompBioMed described within this deliverable.

The BoneDVC workflow involves the processing of very large X-ray micro- and nano-tomography datasets collected at the Diamond Light Source, the UK's national synchrotron science facility. These datasets are typically hundreds of Gigabytes, presenting a challenge for robust and efficient data transfer to an HPC centre for further processing.

Workflow data stored at the Diamond facility may be located either

- on short term online storage, or
- in long term tape archive storage

If archival storage is used, data must first be ‘staged’ to online storage; i.e., reading the data from tape, recovering it to an internet connected server to allow onward transfer.

‘Staging’ of data from the Diamond facility is possible using the ‘TopCat’ [8] web interface, through the following steps;

- User logs in and is presented with a list of their available data

“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No ***675451****“*

- Individual files/directories may be selected and added to the ‘download cart’
- Once all data is in the download cart, staging is requested by opening the cart and selecting the download option

Once staging is requested the data is copied from tape to the data transfer node at the Diamond Facility. This process can take several hours to complete, and the user can opt to receive an email when the data is ready for transfer.

Two options are available for download;

- Exposure using the http protocol with a standard web browser
- Transfer using GridFTP (that uses parallel ftp) with the Globus [9] system for authentication

For transfer to the UK Research Data Facility [10] (UK-RDF) attached to the Archer HPC, GridFTP was chosen due to its improved speed and command line access without requiring web browser access on the HPC platform. GridFTP handles all aspects of file transfer, managing authentication at both source and destination endpoints and ensures the transfer completes successfully and securely. This includes automatic retries, data checksumming and encrypted transfer. Data transfer is requested by the user from a client machine and all authentication to the Diamond and EPCC systems was performed through the Globus API.

GridFTP transfer can be achieved in one of two ways:

- manually using the Globus web interface (see the Globus Manual [11]). Log-in to Globus online allows authentication to Globus endpoints at the Diamond and RDF. The filesystems may be explored and transfer of files between the two endpoints requested.
- in an automated fashion using the API interface to Globus. A Python scripted interface to the Globus System employs the Globus-SDK [12] with additional code to provide configuration of automated transfers. Scripting allows Globus file transfers to be included in any automated workflow and enables queuing of multiple transfers more efficiently than using a graphical approach.

Benchmarking of transfer rates between the Diamond and RDF endpoints was undertaken with datasets of hundreds of gigabytes. This provided rates in the region of 400MB/s for transfers with parallelism=8, (each parallel thread provides ~ 50MB/s).

These developments have delivered a robust, scriptable, solution to data transfer for the BoneDVC workflow which provides transfer rates of the order 40 minutes/ TB.

In addition to the transfer of initial imaging data from the Diamond facility to the Archer HPC facility the same Globus transfer can be used to return processed data to the user following the execution of HPC elements of the workflow. The positioning of the data transfer process within the overall computational workflow is shown in Figure 9 below.

*“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No **675451**“*

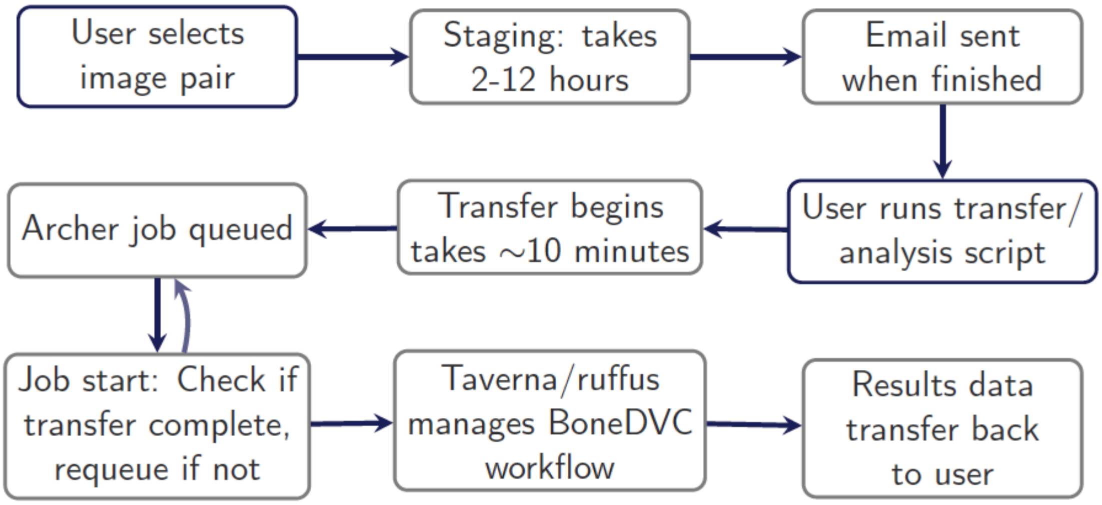
User selects image pair
Staging: takes 2-12 hours
Email sent when finished
User runs transfer/ analysis script
Transfer begins takes ~10 minutes
Archer job queued
Job start: Check if transfer complete, requeue if not
Taverna/ruffus manages BoneDVC workflow
Results data transfer back to user

Figure 9: Use of data staging and transfer process to provide image pair to initiate HPC workflow on Archer HPC facility and to return processed data to user following HPC execution.

9 Best Practices Guidelines

The Archer data transfer guide (see http://www.archer.ac.uk/documentation/data-management/transfers.php) provides best practice guidelines on data transfer to and from the UK-RDF service. It describes pre-transfer strategies such as data archiving to prepare data for best possible transfer speed. It describes transfers using tools such as scp and rsync. Transfer tools such as “Globus Online” and “GridFTP” are described too.

SURFsara Data Archive user information (https://userinfo.surfsara.nl/systems/data-archive/usage) provides guidelines for optimal storage and archiving of data. The guidelines describe the minimum file size for optimal archiving, usage of internal archiving tool such as dmftar, and high performance data transfer to/from the archive using protocols such as HPN-SSH and GridFTP.

From EUDAT, a “data management plans best practices and case study” is available from https://www.slideshare.net/EUDAT/data-management-plans-eudat-best-practices-and-case-study

10 Conclusions

A list of suitable data services available to the CompBioMed users from EUDAT CDI, SURFsara and EPCC were described. Such users should consider these services when creating a data management plan for their particular research work.

This deliverable highlighted the work done in deploying an object storage to meet the requirements highlighted in CompBioMed deliverables D5.2 and D1.3 for the increasing demand for large data storage. This storage service provides over 2.5 petabytes of storage, exceeding the anticipated CompBioMed data storage requirement. Tests carried out on this object storage show that it can meet the CompBioMed requirement for transferring and managing large number of small files as well as for single large files (> 1GB). The PolNet application was adapted

*“This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No **675451**“*

to use this object storage and so shows one of many ways that biomedical applications could use the large reliable storage capacities provided by object storage systems.

Description of data transfer tools that could satisfy various requirements identified in CompBioMed were provided. Use of GridFTP within the Diamond Light Source Pilot (see section 8.4.1) showed how this particular tool could be used for large data transfers, making the effective use of the available bandwidth. Third party tools that use Amazon S3 API (i.e., COSBench and Benchio) were used to perform data transfers to and from the EPCC object storage. As shown in the testing performed between user machines and the EPCC object storage, it too can reach over many hundreds of MB/s using parallel data channels. For effective bandwidth usage, it is necessary to transfer data in parallel using multiple parallel workers (i.e., using multiple parallel data channels). In many cases, the users themselves would not have to choose whether to use parallel workers or not – many S3 clients (although not all) will automatically use parallel workers, and so the user only has to make sure they are using a suitably full-featured Amazon S3 client.

The optimisation work carried out using B2SAFE for data transfer and replication to geographically distributed data storages that are close to HPC resources (i.e., BSC, EPCC and SURFsara) was described. Use of this and other EUDAT CDI services provide many benefits to the community and should be strongly considered when creating a data management plan.

The optimisation work carried out as part of this deliverable, wherever possible, used applications that are in the CompBioMed Software Hub. For example, B2SAFE data replication optimisation work described herein used Alya application data for testing purpose and object storage optimisation work described in here used PolNet application data for various tests.

11 Bibliography

[1] "D5.2 Report on Computing and Data Needs of the Biomedical Community," [Online]. Available: https://www.compbiomed.eu/wp-content/uploads/2017/03/D5.2_ReportonComputingandDataNeedsoftheBiomedicalCommunity_SARA_V1.0.pdf.

[2] "D1.3 Data Management Plan," [Online]. Available: https://www.compbiomed.eu/wp-content/uploads/2017/03/D1.3_DataManagementPlan_CBK_v1.3.pdf.

[3] "EUDAT CDI," [Online]. Available: https://www.eudat.eu/eudat-collaborative-data-infrastructure-cdi.

[4] "Object Store Large Quantities of Data," [Online]. Available: https://www.surf.nl/en/object-store-store-large-quantities-of-data.

[5] "ELEM Bio," [Online]. Available: http://www.elem.bio/.

[6] "EOSC-Hub," [Online]. Available: https://www.eosc-hub.eu/.

[7] "Data Policy Management," [Online]. Available: https://www.eudat.eu/news/a-new-feature-for-b2safe-the-data-policy-manager-dpm-tool.

[8] "Topcat Diamond," [Online]. Available: https://topcat.diamond.ac.uk/. [Accessed 1 August 2018].

[9] "Globus," 25 June 2017. [Online]. Available: https://www.globus.org/. . [Accessed 1 February 2018].

[10] "ARCHER » UK Research Data Facility (UK-RDF) Guide.".

[11] "How To Log In and Transfer Files with Globus - Globus Docs," [Online]. Available: https://docs.globus.org/how-to/get-started.. [Accessed 1 February 2018].

[12] "Globus SDK for Python," [Online]. Available: http://globus-sdk-python.readthedocs.io/en/stable/.. [Accessed 1 February 2018].

[13] "ARCHER," [Online]. Available: http://www.archer.ac.uk/. . [Accessed 1 February 2018].

[14] "ELEM Bio," [Online]. Available: http://www.elem.bio/.

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No **675451**"*