Grant agreement no. 675451

CompBioMed

Research and Innovation Action

H2020-EINFRA-2015-1

Topic: Centres of Excellence for Computing Applications

D6.1 Report on existing solutions in support of biomedical applications

Work Package:	6	
Due date of deliverable:	Month 12	
Actual submission date:	29 / September / 2017	
Start date of project:	October, 01 2016	Duration: 36 months

Lead beneficiary for this deliverable: *USFD*

Contributors: *UCL, UPF, UvA, BSC, UNIGE*

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	

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

Disclaimer

The content of this deliverable does not reflect the official opinion of the European Union. Responsibility for the information and views expressed herein lies entirely with the author(s).

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**"*

1 Version Log

Version	Date	Released by	Nature of Change
V1.0	27/06/2017	Alberto Marzo	First Draft/Layout
V1.1	14/07/2017	Alberto Marzo	Extension of First Draft, Introduction
V1.2	20/07/2017	Marco Viceconti	Revision first draft
V1.3	10/08/2017	Alberto Marzo	Integrated contributions from USFD, UCL, UPF, UvA, BSC. End-user engagement plan.
V1.4	11/08/2017	Marco Viceconti	Final revision.
V1.5	23/08/2017	Andrew Narracott	Minor edits prior to internal review
V1.6	11/09/2017	Alberto Marzo	Addressed reviewers' comments
V1.7	15/09/2017	Alberto Marzo	Addressed reviewers' comments with contribution from co-authors

2 Contributors

Name	Institution	Role
Alberto Marzo	USFD	Author
Marco Viceconti	USFD	Author
Shannon Li	USFD	Author
Andrew Narracott	USFD	Author
Alessandro Melis	USFD	Author
Stefan Zasada	UCL	Author
Robin Richardson	UCL	Author
Gabor Zavodszky	UvA	Author
Adria Perez	UPF	Author

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

Jazmín Aguado-Sierra	BSC	Author
Gary Tresadern	Janssen	Reviewer
Toni Collis	EPCC	Reviewer
Peter V 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
CoE	Centre of Excellence
HPC	High Performance Computing
SME	Small and Medium Enterprises
WP	Work Package
PM	Project Month
CBM	CompBioMed

4 Executive Summary

The purpose of this deliverable is to describe the existing solutions adopted in CompBioMed within the area of *in silico* Medicine, their potential end-users and the way they may benefit from such solutions, and the plan that the Centre of Excellence will adopt to engage with the end-users within and beyond the project network, gather their requirements and use these to adapt existing solutions to ultimately maximise their uptake in the users' community. The deliverable provides a description of 11 applications from the areas of Cardiovascular and Respiratory, Neuromusculoskeletal, and Molecular Medicine, together with a precise taxonomy of end-users, provision modalities and motivations that will provide structure to user needs and the planning of the deep-track activities.

*"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 main objectives of the CompBioMed Centre of Excellence (CoE) is to engage with a range of end-users across the entire healthcare value chain, from healthcare providers to pharmaceutical and medical device manufacturers, as well as researchers and HPC system providers, to ultimately exploit high performance computing and build a community of users that will benefit from the CompBioMed services. In order to do this, we have to prepare and expose existing supporting tools, workflows, services and datasets to a large variety of end-users, gauge their interest and adapt these into vertical solutions to specific end-user requirements making them applicable to a wider range of biomedical applications, while attuning them to run effectively in HPC systems. The primary aim of this document, *D6.1 "Report on existing solutions in support of biomedical applications",* is to provide a description of the existing solutions adopted in the CoE and their status with respect to HPC deployment. The document also illustrates the end-user categories that we envisage will benefit from these solutions and will be part of our community, the envisaged provision modalities through which end-users will access our services, the main motivations to combine solutions with HPC for the different end-user categories, and a plan for end-user engagement.

This document reports the results of activities conducted during months M1-M12 in WP6 task T6.1:

Task 6.1: Expose and Publish Existing Solutions (M1-M18) [Fast Track]

Leader: USFD (4 PM);

Partners: UCL (2 PM), UPF (2), UvA (2), UNIGE (2), BSC (2)

In this task, we will provide easy user access to existing end-user solutions (e.g., workflows, supporting tools, services and datasets), both commercial and non-commercial, to increase their uptake in a wider range of biomedical applications. We will draw on the software repositories created as part of Task 5.6, and will ensure that everything required for each end-user solution is properly hosted. At the end of the first year we will produce a public document that details all end-user HPC-enabled solutions available at that point (D6.1); from then we will advertise the available support solutions to the wider research and clinical community (in collaboration with WP3), and create an online platform to help academic, industrial, and clinical researchers find the tools, services and datasets they need.

This report will be used as a reference for the dissemination activities of WP3 *"Training and Dissemination"* in task 3.1 *"Production of a Dissemination Action Plan"* and task 3.2 *"Maintaining the CompBioMed Online Presence";* it will also provide input to the deliverable D6.4 *"Report on selected emerging use cases for existing solutions"*.

6 CompBioMed end-users

There are four general categories of end-users to be served by the CompBioMed CoE:

1) Researchers
2) Clinicians
3) Innovators
4) Educators

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

The first category refers to users who will use our HPC-enabled solutions to pursue Research and Development objectives, both within not-for-profit and for-profit organisations. Probably the best way to taxonomise research users in this category is by the level of "alphabetisation" with HPC systems:

1.a) Researchers in Computational Science & Engineering, numerical analysts, software developers, who are interested in the development, and optimisation of HPC codes;

1.b) Computational bioengineers, biophysicists and biochemists, who have basic understanding of software and hardware, and are willing to learn the use of some specific tools in return for a greater flexibility and configurability;

1.c) Life-science researchers, experimental bioengineers, computational chemists, biophysicists and biochemists, who are only interested in using ready-to-use tools for applications possibly in an industry setting, and need to minimise the time spent learning their usage.

Clinical users represent a continuum from fundamental research to clinical practice. Here for convenience we separate the category in:

2.a) Clinical researchers who contribute to the development or the validation of a predictive model;

2.b) Clinical researchers who use validated predictive models or their outputs to support their clinical research, for example the testing of disease mechanism hypotheses;

2.c) Clinical researchers who use HPC-enabled solutions to support, augment, and supplement clinical trials of new interventions not associated to a commercial product.

Innovators use HPC-enabled solutions to develop new products and services to be widely commercialised. In this case the best taxonomy derives from the different phases of biomedical innovation:

3.a) Innovators who work in the design and optimisation of new products;

3.b) Innovators who work in the safety and efficacy assessment of new products, and in the related regulatory activities;

3.c) Innovators who work toward the development / deployment of commercial computational medicine solutions.

The last category, educators, indicates a broad category of university lecturers who teach at various levels of higher education, and in various disciplines related to life science and medicine, who may use HPC-enabled solutions to improve their ability to teach the role that these methods and technologies can play in the handling of complexity typically associated with life sciences and medicine.

It is important to notice that from a user perspective the metrics of success vary depending on the end-user goal:

- HPC for researchers
  - Low overhead for one-off execution: in some cases researchers need to run one very large model only once; thus, an important metric is how small the overhead is to port that application and execute it within an HPC environment;
  - Re-usable computational frameworks: in other cases, researchers are interested in using HPC-enabled solutions as a recurrent tool in their research; here the

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

availability of validated, well documented and widely used HPC-enabled frameworks targeting specific research areas are essential;

- HPC for Industry
    - Validated workflows: confidence and reproducibility are key concepts in the emerging regulatory science for modelling & simulation in biomedicine; all HPC-enabled solutions that can offer out-of-the-box confidence (through validation, extensive use, benchmarking, or certification) are preferable;
    - "My data your model": industrial research and innovation is mostly highly confidential; delivery models that allow industrial users to run simulations on their sensitive data within a secure sandbox are clearly preferable;
- HPC for clinical trials
    - Backend support: clinical trials are first and foremost logistic and organisation nightmares; any HPC-enabled solution to be used within a clinical trial must be packaged with extensive backend support, which ensures high-reliability, full users support, etc.
    - Automated data flow: for phase III multicentric clinical trials the full automation of the data flow from and to the clinical trial databases is vital;
- HPC for computational medicine services
    - Need for service-level agreement (SLA): companies that plan to sell services that are partially or fully based on HPC-enabled solutions provided by a third party need well-crafted Service-Level Agreements that shield them from the risks associated with the operation of the HPC infrastructure;
    - Cost-effectiveness: at the same the total cost of service plays an important role for companies that act as value-added re-sellers of HPC-enabled solutions.

One additional requirement that every type of end-user group shares is the need for training. But training needs to be tailored to the needs of each group of end-users, and will depend upon where the prospective users are in their professional trajectory; this aspect is being addressed in WP3, and will not be discussed here any further.

7 Provision modalities for end-users

So far, our horizon scan has identified a limited number of modalities through which HPC-enabled solutions can be made available to end-users:

- *Source Code:* the software tool is made publicly available for download (for example via GitHub), together with manuals and tutorials that help in the installation, configuration and use of the solution directly by the end user on the system of their choice;

- *Direct (or binary)*: the solution is installed, configured, and optimised for one or more specific HPC targets. The end-users must gain an account and CPU-time allocation of that system, and then they can use the solution directly. For research users, the solution is accessed by remote login to the HPC system; for clinical and industrial users, specific remote batch submission and results retrieval can be devised;

- *Software as a service (SaaS):* the solution is accessed through an easy to use interface, for example a web portal, which acts as a front-end to the computational medicine solution that runs on HPC compute nodes (typically clouds);

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

- *Indirect:* the solution involves some elements of specialised consulting, and thus its access is provided indirectly, in the sense that the user sends the input data to the provider who runs the solution for them and returns the results to the end-user.

8 Motivations to port computational medicine applications to HPC

What are the motivations to use HPC-enabled computational medicine solutions? From our horizon scan, the most common motivations that force our users to use an HPC resource instead of a conventional platform are:

- Run full order model to:
  - Solve models whose computational complexity cannot be reduced (unreducible models)
  - Validate reduced-order models, which requires you to run a full-order model once
- Scale full order model to:
  - Test convergence, which requires you runthe model once at higher resolution
  - Model larger space-time regions, while preserving the resolution
  - Provide actionable predictions - results which can be fast enough to lead to key decision making such as interventions, drug discovery, etc.
- Run model repeatedly to:
  - Perform uncertainty quantification over computational expensive models
  - Inform surrogate model, which require to run a model over the entire input parameter space of admissible values
- Combinatorial explosion in:
  - Multiscale models
  - Strongly coupled models

9 End-user solutions

9.1 Overview

In this section, we provide a description of each solution (tools, workflows, services and datasets) adopted in CompBioMed together with an update of their porting status to an HPC system, their current access mechanisms, barriers encountered, and how these solutions are being adapted to fully exploit HPC systems. The broad range of solutions presented below are clustered to ease their presentation in three categories: cardiovascular and respiratory, neuromusculoskeletal, and molecular medicine.

9.2 Cardiovascular and respiratory solutions

9.2.1 Cardiovascular applications with Palabos

Palabos, developed at the University of Geneva, is an open-source software for the simulation of fluid flow, including complex physical, chemical, or biological interactions. It is used in many domains of applications. Within the CompBioMed project, we produce and maintain a number of Palabos-based numerical simulators able to solve biomedical problems. These include a simulator for blood flow in human arteries, with the following capabilities:

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

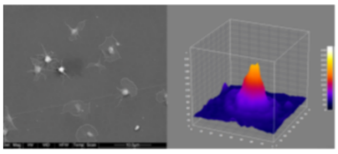
- Automatic simulation setup from a description of artery shapes obtained through numerical imagery;
- Instantiation and simulation of pulsating blood flow;
- Deployment of medical devices, like stents, in the arteries, and analysis of their impact on the blood flow.

The ongoing deep-track research in CompBioMed aims at improving the capabilities of this solver. The research includes:

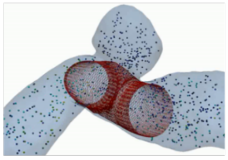
- *Ab initio* modelling of biological processes, such as platelet deposition;
- Formulation of stent models. These allow the effect of a stent on the blood flow to be accounted for through a force model, without needing to fully resolve a stent structure in the simulation;
- Fully resolved model for Red Blood Cells, with massively parallel capabilities, allowing inclusion of billions of Red Blood Cells in the simulations.

These three research areas are represented by the following illustrations.

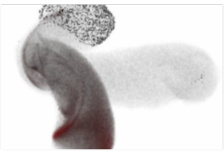
Ab initio modelling of biological processes



Models for medical devices



Inclusion of Red Blood Cells



One of the most striking features of the Palabos software is its massive parallelism. The software uses parallel computers with the order of 10^5 cores, and exhibits an excellent scaling behaviour. This parallel scaling applies not only to the solver itself, but also to the pre-processing stages, including the mesh generator, which in this field is referred to under the name of “voxelizer”.

To encourage collaboration between Palabos-based solvers and other numerical tools of CompBioMed, a stand-alone version of the Palabos’ parallel *voxelizer* was created, which is intended to be used, among others for the HemeLB solver.

9.2.2 HemoCell – High-performance Microscopic Cellular Library

Blood is a complex suspension constituted of various components suspended in plasma. Many of its intriguing properties originate from this cellular nature. HemoCell[1,2] is a parallel computing framework which implements validated mechanical models for red blood cells and is capable of reproducing the emergent transport characteristics of such a complex cellular system. With the recent advancement of micro-medical devices modelling the small-scale behaviour gains more importance. Accurate modelling of blood flow related phenomena on this scale requires a description of the dynamics at the level of individual cells. This, however, presents several computational challenges that can only be addressed by high performance computing.

To facilitate the deployment of HemoCell for end-user solutions, the build system has been rewritten to make it compatible with a range of build environments available in various supercomputing centres. Using this new build method HemoCell was successfully built on the following HPC systems: Lisa, (SURFsara, The Netherlands); Cartesius, (SURFsara, The Netherlands); SuperMUC (LRZ, Germany), Superman (BME, Hungary), Lomonosov-2 (Evrone, Russia).

To exploit these HPC systems with optimal efficiency, the data structures and the communication model of the framework has been re-implemented resulting in a more than 10-times performance increase across our every test case.

We also made improvements to extend end-user applicability by including a dataset containing the mechanical properties of several cell-types[2] (such as healthy and malaria infected red blood cells, platelets, white blood cells) queried from available literature. Furthermore, we extended the pre- and post-processing facilities which helps to set the computations up quickly and to visualise and evaluate the results. Finally, we set up a website for the framework (http://www.hemocell.eu).

9.2.3 HemeLB stent design and magnetic drug targeting pipeline

HemeLB, developed at UCL, is a bespoke lattice-Boltzmann based fluid solver designed to efficiently model cranial bloodflow. This solution simulates blood flows through a stent (or other flow diverting devices) deployed in a patient's cerebral artery[3]. The solution enables the evaluation of different stent designs (surface patterns) and how they affect the viscous 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 drug-coated particles, to be simulated and estimates made as to where they might statistically end up.

More technically, the pipeline takes as input a stereo lithography (STL) file of the surface geometry of the patient's artery, generally obtained via segmentation of Digital Imaging and Communications in Medicine (DICOM) images from a CT-scan. It also requires input data on the velocity-time profile of fluid flow at each of the inlets to the simulated region, and the start and end locations of the deployed stent, and a binary image of the stent design patter. The HemeLB setup tool voxelises the volume bounded by the input STL at a user-specified 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 vectorial fields, or 3D shots of wall shear stress distribution in the geometry using ParaView visualisation software.

HemeLB (and associate software of the pipeline) is available on an open-source licence, and already runs on major HPC facilities in Europe (e.g. EPCC ARCHER, LRZ SuperMUC, PSNC Prometheus). Major development work has been undertaken to reduce HemeLB's memory footprint, allowing efficient exploitation of HPC systems. For example, recent benchmarking runs have been executed on up to 96,000 cores on ARCHER, showing good strong scaling performance for even relatively small (777 million fluid sites) geometries. Current work is focussing on increasing this to ~200,000 cores.

9.2.4 ALYA - Multi-scale, multi-physics cardiovascular and respiratory simulations

Alya is a multi-scale, multiphysics, HPC native code used to solve complex physiological models to understand the cardiovascular system in health, disease and during its interaction with medical implants or invasive therapeutic devices. As such, the code has been developed from

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

its infancy to run using HPC facilities[4]. It has been developed to solve electrophysiology, solid mechanics, fluid flow and contact problems; tightly and loosely coupled. Alya provides the feasibility of coupling it to other simulation codes to solve complex coupled problems. Coupling between HPC codes is a possibility that has not yet been required for cardiovascular or respiratory simulations; therefore, this is not yet part of complex workflows. On the other hand, post-processing of data with machine learning techniques, optimization, uncertainty quantification and sensitivity analyses is a work in progress. Model parameterization for patient-specific applications is a problem that can be tackled by the use of a complex workflow that involves simulation, optimization and machine learning. The pipeline for the use of optimisation tools like Dakota (https://dakota.sandia.gov/) has been developed at BSC and it is being employed for electrophysiological parameter calibration to experimental data. Work is currently underway regarding this particular workflow. Complex workflows for cardiovascular and respiratory solutions are being developed and will be addressed within WP6 in the Deep track phase of the project.

A production release of Alya is available from the software hub of CompBioMed and it has been deployed and is maintained in the following HPC facilities, partners of CompBioMed: Cartessius, (SURFsara, The Netherlands), Archer (EPCC, UK), and Marenostrum 4 (BSC, Spain). Accessibility to the software is currently granted by contacting Mariano Vázquez or Guillaume Houzeaux, from BSC, with no usage restrictions for CompBioMed partners.

9.2.5 OpenBF

OpenBF is a numerical model that simulates the physics of pressure and flow-rate waves propagation across the elastic vessels of a typical arterial network. The model predicts pressure and volumetric flow values along the cardiac cycle (waveforms) at several locations along each vessel of a network that currently includes 78 arteries of the systemic and cerebral circulation. This Navier-Stokes-based 1D model[5] is fully validated, and its numerical predictions are computed by means of the MUSCL explicit finite volume scheme. OpenBF is written in Julia and it has been released as open-source under a LGPL-2.1 license (https://github.com/INSIGNEO/openBF). The code is well documented and supported by installation guidelines and tutorials to ultimately maximise end-users' uptake.

As the current openBF implementation is serial and does not exploit multi-core architectures, within CompBioMed, the code has been deployed and tested on the SURFsara cloud system. In this environment, OpenBF will be used to populate a database of waveform datasets to be used as boundary conditions by the higher-dimensionality cardiovascular models in the consortium. The waveforms contained in the database will be calculated by means of Monte Carlo analysis over typical physiological ranges for all the model parameters. The resulting database, which will be made available through the CompBioMed Centre of Excellence, will be annotated and filtered to retain only realistic waveforms towards the development of a database representing a virtual population.

Future developments of OpenBF includes extension of the code that will allow strong-coupling to higher-dimensionality codes, and the exploration and testing of clinically-relevant hypotheses where cardiovascular disease might influence waveforms, and inform future development of diagnostic technologies.

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

9.3 Neuromusculoskeletal solutions

9.3.1 BoneDVC

The measurement of the biomechanical strain induced in bone tissue at the scale 10-50 microns is essential to understand the mechanoregulation of bone cells. Unfortunately, no strain sensor exists that can operate at such small scale; the only viable option is to perform two microCT scans, one of the undeformed specimen, and one of the same specimen under a static loading, and then use elastic registration techniques to estimate how each material point in the undeformed 3D image moves to different spatial coordinates in the deformed 3D image. This technique is called Digital Volume Correlation. Its application to bone tissue is particularly challenging, because bone is stiffer than most other tissues, and thus the displacements involved are much smaller. USFD has developed a new DVC algorithm for the analysis of bone tissue (Called BoneDVC) that is currently the most accurate among those reported in the literature.

BoneDVC is a computational medicine workflow aimed at the calculation of the strain field induced by staged compression in a specimen of bone tissue. The workflow employs the Digital Volume Correlation (DVC) technique to calculate the displacement field given two sets of microCT images (reference and compressed states, respectively). Subsequently, the strain field is calculated by means of differentiation of the displacement field.

The code was originally developed to analyse data generated by low-resolution desktop microCT systems, and did not pose particular computational challenges. However, the research community now wants to analyse, with the same method much larger datasets, generated with newer high-resolution NanoCT systems, or with Synchrotron-light CT, such as that provided by the Diamond Light Source. Such datasets cannot be analysed on conventional platforms, and require an HPC version of BoneDVC.

BoneDVC employs ShIRT (Sheffield Image Registration Toolkit, written in C) to compute the displacement filed, and a series of Matlab codes to compute the strain field. The different codes are orchestrated by a Taverna workflow and the HPC execution is done through a series of Python scripts. BoneDVC has been deployed on ARCHER (EPCC) and ShARC (USFD)[6]. On ARCHER, where Matlab is not available, the Matlab scripts have been compiled and run through the Matlab Runtime Environment (mcc), which is license-free.

BoneDVC is written in serial code, and therefore does not scale on an HPC architecture; a code analysis was conducted, which confirmed the computational bottleneck is the ShIRT code, which needs to be parallelised; this will be pursued as part of the deep-track activities.

9.3.2 CT2S

CT2S, or ‘CT to Strength’, is a workflow to calculate femoral strength of a specific patient based on individualised bone geometry and mechanical properties. The predicted bone strength can then be used by clinical researchers as a biomarker in clinical trials on any intervention (pharmacological, physical, or surgical, including regenerative medicine) aimed to strengthen the skeleton of patients affected by a variety of pathologies; our recent study shows that CT2S could reduce the size of the clinical cohort of up to 50%, with respect to the most commonly used alternative, which is to estimate strength from a correlated property, femoral neck mineral density. Our group is also exploring the use of CT2S in the clinical routine, as a specialised prognostic tool for a subgroup of osteoporosis patients which are notoriously difficult to prognosticate; in this second scenario, there is potentially an element of Urgent Computing, as ideally the patient-specific prediction should be returned to the requesting

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

clinical specialist with the duration of the visit (20 minutes); currently we are able to ensure a turnaround of 48 hours. This poses issues of full automation of the modelling workflow, but also the use of the HPC resource within an Urgent Computing framework.

The CT2S workflow takes input as Quantitative Computed Tomography (QCT), a fairly standard medical imaging procedure; the CT2S imaging protocol has been optimised to minimise the x-ray radiation exposure for the patient, while ensuring the quality that is necessary for the subsequent simulation. Currently, the workflow is not fully automated because the segmentation of the bone from the QCT images requires the intervention of a trained operator. Once the QCT is segmented the 3D image and the 3D surface are sent to a full automated workflow where a) the 3D surface is automatically meshed into a 10-node parabolic tetrahedron finite element mesh of excellent quality, using a commercial tool (ICEM v15, Ansys, USA); b) the resulting mesh is back-projected onto the 3D image and the calibrated attenuation coefficients that are represented as voxel value in the QCT data are used to assign to each finite element a different set of elastic constants related to the local degree of mineralisation, using a freeware tool (http://www.bonemat.org). The resulting finite element model is then solved within a Monte Carlo scheme to calculate the minimum strength as the direction of fall is randomly sampled, the representative contact boundary conditions are generated automatically. The Monte Carlo and the minimum strength post-processing calculations are implemented in a Matlab code, as they pose very little computational cost. The solution of the finite element model is done with a commercial code (Ansys Mechanical v15, Ansys, USA). All the components are orchestrated into a single workflow executed by a workflow management system (http://www.taverna.org.uk). Considering this code will soon have to be certified for clinical use, the use of ICEM and Ansys, or other equivalent industrial-strength codes with all necessary certifications, appears mandatory.

Currently, CT2S is exposed to registered users through a dedicated portal (https://ct2s.insigneo.org/). The user uploads the anonymised QCT data of the patient in association to a user-provided Patient ID code, that will be used to report the results, and requests a strength analysis. The system adds the new case to the worklist of an operator, who perform the manual steps, executes the Taverna workflow, inspects the results, and upload it to the portal, which automatically generates a full PDF report that is sent to the requesting user.

Originally these patient-specific models involved the solution of a single finite element model with a few hundred thousand degrees of freedom, and a linear elastic constitutive equation. But in the last years, the team of Prof Viceconti at USFD has demonstrated that a number of improvements could significantly increase the predictive accuracy of these models, including:

a) The inclusion of large displacement and material non linearities, the latter especially for the analysis of vertebral fractures, where considerable plasticity is observed;
b) The inclusion of large-sliding frictional contact non-linearity to better model the complex boundary conditions acting on the femur during a sideway fall event;
c) The need to run the same model under a large range of possible impact directions, in order to compute the *minimum side-fall strength*, which our studies confirmed to be the best predictor of the risk of femoral fracture in osteoporotic patients.

Currently, a single CT2S case requires 130 core-hours using the serial version of the software on one core. Using the MPI version of Ansys running on the ShARC system that scales well up to 16 cores thanks to a more efficient Preconditioned Conjugate Gradient (PCG) iterative equation, plus the parallelisation of Monte Carlo has reduced the runtime nearly 1.5 hours (2 hours including the queue waiting time), a speedup of 86 on 688 cores. In addition, to improve

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

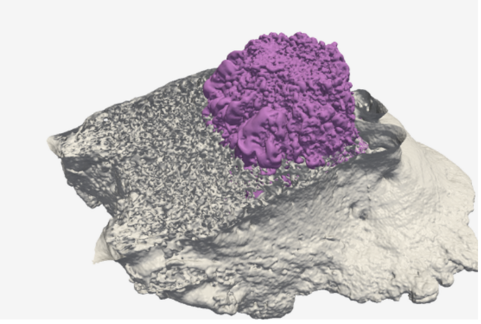
the time to solution from submission we can use priority queuing we are able to reduce this further by removing the majority of the queue time. This is particularly important to our work where time to solution is a key factor, rather than just the compute time and cost.

9.3.3 Neuromusculoskeletal applications with Palabos

Vertebroplasty procedures are performed on patients who due to osteoporosis, a cancer metastatic lesion, or an accident, have one or more collapsed vertebral bodies. A polymeric material known as bone cement is injected into collapsed vertebral body to re-expand it to its normal size, remove the pain, and prevent any further collapse of that vertebrae)

A numerical simulation of vertebroplasty cement injection based on Palabos (described in the section on cardiovascular applications), is maintained for the purposes of the CompBioMed project. During the vertebroplasty procedure bone cement is injected through a small hole in the skin into a fractured vertebra. Patient-specific computer simulation is an ideal tool to assess the risks of an intervention, and to provide a surgeon with recommendations for the amount of cement to be injected, and other parameters.

The Palabos-based solver allows the set-up of a simulation for a patient-specific vertebral bone obtained from medical imagery, and predict the benefits and limitations of a corresponding surgical intervention. The result of such a simulation is shown in the following image. This shows that in this case, a given quantity of injected cement remains within the boundaries of the vertebral bone and does not overflow and damage vital body parts. Such simulations are instrumental to guide the doctor to choose the proper amount of cement.



This type of simulation is patient-specific, and the execution speed of the simulation is therefore critical. The simulations receive substantial speed up from massive parallelism, and from a dynamic load balancing strategy that repartition the domain by following the front of the injected cement at selected moments in time.

9.4 Molecular medicine solutions

9.4.1 BAC - Automated Binding Affinity Calculations

The Binding Affinity Calculator (BAC) is a decision support tool which uses molecular level computer simulation to reliably predict the binding affinities (free energies) of molecules with target proteins, and therefore to identify those most likely to bind to the protein. BAC has been built to integrate and automate the multi-step process of model building, simulation and data analysis for molecular level drug-receptor interactions[7,8]. It constitutes a sophisticated computational pipeline built from selected software tools and services, and which relies on access to a range of computational resources. Ultimately, one important use is a more

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

accurate computational design of candidate drug molecules. In fact, BAC is currently in use with a number of pharma companies.

BAC depends on the ability to perform hundreds of separate parallel simulations on a high-performance computing platform, each of which can require 50-200 cores depending on the system. The BAC workflow automates much of the complexity of running and marshalling these simulations, and collecting and analysing data.

BAC constitutes a workflow of operations encompassing several different binaries along with data creation and transformation scripts. It can be deployed in multiple application scenarios, but is principally used as either a direct 'application' run by a researcher on an HPC platform (which depends on significant HPC experience in order to use successfully), or more latterly in a Software as a Service model, where the deployment is taken care of on a user's behalf, and they actually run BAC via a web interface.

UF-BAC is a web portal based interface to the binding affinity calculator (BAC), which allows a user to build models of molecule-compound binding, and execute and analyse multi-replica molecular dynamics (MD) simulations using the model. The binding affinity may be calculated by ESMACS or TIES methods. UF-BAC enables BAC to be run via a Software as a Service model, hiding from the user the complexities of the command line tools used to build models, execute them on compute resources, and analyse the results. UF-BAC is the interface to BAC and is required to run the BAC application.

9.4.2 PlayMolecule

PlayMolecule is an interactive[9] platform that intends to gather different solutions for Molecular research, and integrate them in an intuitive, easy-to-use interface. PlayMolecule currently hosts a group of web applications to perform different activities, such as a pocket detection tool (DeepSite) or a protein preparation tool for MD simulations (proteinPrepare). The web uses a similar display for every application to facilitate the user experience with the web applications. The main purpose of PlayMolecule is to deliver different end-user solutions under the same interface style to make the web applications more interactive, connected, accessible, and user-friendly. PlayMolecule can be accessed through www.playmolecule.org where users get access to our GPU-based servers to submit jobs through the web applications. Users can use a guest account or create their own account for a private space to save their results.

The PlayMolecule platform is a web application that fully leverages modern web technologies such as WebGL and the Material Design guidelines. Hence, it enables a rich and dynamic interaction, once only attainable by desktop applications, with the advantages of web technologies such as platform independence, being installation- and update-free, and so on. These advantages, together with steady improvements in web technologies and connectivity, suggest that rich web applications may become pivotal in academia and the pharmaceutical industry alike.

9.4.3 HTMD

HTMD (High-Throughput Molecular Dynamics) is a programmable, extensible platform written in Python. It provides a complete workspace for simulation-based discovery through molecular simulations while aiming to solve the data generation and analysis problem as well as increase reproducibility. With HTMD, we intend to integrate the whole workflow of molecular simulation-based discovery in a single environment while abstracting unnecessary technical details. This reduces preventable errors in the workflow, increases reproducibility of molecular

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

dynamics experiments, allows for the manipulation of large amounts of simulation data (order of terabytes), and opens the way for biologists and medicinal chemists to utilize simulations while focusing on the real biological problems they are trying to solve.

HTMD (https://www.htmd.org) extends the Python programming language with functions and classes to handle molecular systems at different levels while abstracting implementation details and best-practice knowledge. Python is a scripting language that enjoys widespread usage in the scientific community and thus provides an ideal platform on which to develop and distribute HTMD. HTMD's functionalities span from molecular structure manipulation to visualization, preparing and executing molecular simulations on different computing resources, and data analysis, including the use of Markov state models (MSMs) to identify slow events, kinetic rates, affinities, and pathways. HTMD provides the user with an integrated platform for in silico molecular simulation discovery. Its functionalities range from molecular structure manipulation to system building, docking, MD simulations, simulation management, clustering, Markov models, and adaptive sampling. HTMD can be used from any python interpreter in the form of self-executing scripts as well as interactively using ipython or jupyter notebooks. The jupyter notebooks allow a user to combine code, documentation, and figures in one document, thus integrating a whole experiment, setup, and report in a single file, which can help increase the reproducibility of experiments. Additionally, notebooks provide the possibility for full remote execution of HTMD via a server (e.g., Amazon EC2) and a browser. The HTMD software, documentation, tutorials, and examples are available at www.htmd.org, where it can be downloaded for free by academic users.

10 A strategy for end-user engagement and community building

The initial exemplar solutions described above will be exposed to the different end-users' categories by adopting a *push-pull* promotional strategy.

In a first phase (*pull*, M01-M14), we will expose existing solutions to existing core and associated partners' users, that will have access to a list of solutions (this list is currently being populated) that includes a lay and technical description and their current access mechanisms. We will then compile a table that will map our solutions to potential end-users and contact users to assess service requirements. This activity will inform the future development of each solution during the deep track phase of the project.

In a second phase (*push*, M14-M19), we will expose the same list to external end-users in the wider communities leveraging international programmes and their networks (Insigneo Institute for *in silico* Medicine, The Virtual Physiological Human Institute, the Avicenna Alliance).

A first set of results from this process will populate deliverable D6.4 “Report on selected emerging use cases for existing solutions” which will update this deliverable with the solutions available at the end of the second year of CompBioMed. However, the end-user engagement process will continue until the end of the project, in coordination with the work on training and dissemination done in WP3.

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

11 Data Management requirements

A detailed review of the community requirements for data management has been undertaken as part of Work Package 5 and is provided in *D5.2 Report on computing and data needs of the biomedical community.* This describes both the data types associated with CompBioMed applications and the general requirements of the biomedical community, typically characterized by the size of the underlying problem to be addressed by the computational workflow. D5.2 also reviews data management requirements in terms of data access and retrieval, handling of metadata and identifiers.

Data Storage requirements and infrastructure have been considered in terms of disk space usage and long term storage needs, for both non-simulated (generally used to build the model) and simulated data (generated from computational models). The typical total output data volume ranges between a few Gigabytes and tens of Terabytes for CompBioMed applications.

Data storage services provided by the HPC centres present in CompBioMed provide underlying infrastructure for Data Management (Table 8, D5.2) with additional functionality provided by EUDAT services (www.eudat.eu), supported by CompBioMed partners UCL and SURFsara. Additional information about the policies that will be adopted and the plans for the medium and long term storage of CompBioMed data can be found in the data management plan D1.3 and is reported in the deliverable “D6.2 Deployment of project informatics platform”.

In the context of Workpackage 6 it is important that Data Management solutions address issues raised by both workflow-specific and end-user-specific requirements. As a result, Work Package 6 effort will be used to inform the development of underlying Data Management technology within Work Package 5 through the end-user engagement and community building strategy described above. This will be achieved through specific attention to data access requirements as part of the end-user assessment of service requirements and will be reported in D6.4 “Report on selected emerging use cases for existing solutions”.

12 Conclusions

The 11 solutions that were made available on one or more of the accessible HPC systems in this fast-track initial phase are excellent examples of the breadth and depth required in computational medicine, and that makes a European Centre of Excellence on HPC in Computational medicine indispensable.

This initial work has also provided us with a taxonomy of end-users, provision modalities and motivations that will provide structure to the needs analysis and the planning of the deep track developments.

The HPC territory that emerges from these early solutions is quite different from the traditional HPC of *ab initio* molecular dynamics and quantum chromodynamics codes, characterised by CPU-bounded computational challenges. Computational Medicine requires a variety of tools, where the computational challenge can be CPU, I/O, or memory bound, or more frequently combinations of these. It is time to move from the scenario where we chose the problems that could be addressed with the HPC we had, to a new scenario where we deliver the HPC that is necessary to solve the problems the users have. The specifications that are emerging for the exascale computing initiative from this domain are likely to be disruptive.

There is also a huge data management problem: some of these codes move hundreds of terabytes of data, which need to be transfered to and from the HPC system, handled efficiently

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

in term of I/O, explored (ideally with some form of interactive visualisation), and stored permanently and securely so that specific elements can be easily and quickly extracted later on for reuse.

13 Bibliography / References

1. Zavodszky, G., van Rooij, B., Azizi, V., Alowayyed, S. & Hoekstra, A. Hemocell: a high-performance microscopic cellular library. *Procedia Comput. Sci.* **108,** 159–165 (2017).
2. Závodszky, G., van Rooij, B., Azizi, V. & Hoekstra, A. Cellular Level In-silico Modeling of Blood Rheology with An Improved Material Model for Red Blood Cells. *Front. Physiol.* **8,** 1–14 (2017).
3. Groen, D. *et al.* Analysing and modelling the performance of the HemeLB lattice-Boltzmann simulation environment. *J. Comput. Sci.* **4,** 412–422 (2013).
4. F. Levrero-Florencio, E. Zacur, A. Minchole, M. Vázquez, J. Aguado, G. & Houzeaux, V. Grau, B. R. HPC electromechanical simulations of the human heart: an MRI-based validation. *Front. Physiol.*
5. Melis, A., Clayton, R. H. & Marzo, A. Bayesian sensitivity analysis of a 1D vascular model with Gaussian process emulators. *Int. j. numer. method. biomed. eng.* 1–11 (2017). doi:10.1002/cnm.2882
6. A. Melis, E. Dall'Ara, A. Marzo, M. Vazquez, G. Pringle, M. Viceconti, A. & Consortium, C. HPC for the rest of us: porting real-world research applications to and HPC environment. *J. Comput. Sci.*
7. Wan, S. *et al.* Rapid and Reliable Binding Affinity Prediction of Bromodomain Inhibitors: A Computational Study. *J. Chem. Theory Comput.* **13,** 784–795 (2017).
8. Wan, S. *et al.* Evaluation and Characterization of Trk Kinase Inhibitors for the Treatment of Pain: Reliable Binding Affinity Predictions from Theory and Computation. *J. Chem. Inf. Model.* **57,** 897–909 (2017).
9. Martínez-Rosell, G., Giorgino, T. & De Fabritiis, G. PlayMolecule ProteinPrepare: A Web Application for Protein Preparation for Molecular Dynamics Simulations. *J. Chem. Inf. Model.* **57,** 1511–1516 (2017).

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