BIOENGINEERING MODELING AND COMPUTER SIMULATION
Y. González and M. Cerrolaza (Eds.)

TOWARDS THE MULTISCALE MODELLING OF THE MUSCULOSKELETAL SYSTEM

Marco Viceconti *
Laboratorio di Tecnologia Medica
Istituti Ortopedici Rizzoli
Via di Barbiano 1/10, 40136 Bologna, Italy
Email: viceconti@tecno.ior.it
Web page: http://www.ior.it/tecno/

Fulvia Taddei[1], Serge Van Sint Jan[2], Alberto Leardini[3], Gordon Clapworthy[4], John Domingue[5], Stefania Galizia[5] and Paolo Quadrani[6]
[1] *Laboratorio di Tecnologia Medica, Istituti Ortopedici Rizzoli – Bologna, Italy*
[2] *Dept. of Anatomy, Université Libre de Bruxelles, Belgium*
[3] *Laboratorio di Analisi del Movimento, Istituti Ortopedici Rizzoli – Bologna, Italy*
[4] *Dept. of Computing and Information Systems, University of Bedfordshire – Luton, UK*
[5] *Knowledge Media Institute, Open University – Milton Keynes, UK*
[6] *CINECA Supercomputing Centre – Bologna, Italy*

Abstract. A number of large research initiatives (IUPS Physiome, EuroPhysiome, SimBio, etc.) aim to develop computer models of human physiology that span multiple dimensional and temporal scales. The Living Human Project (LHP) is a three-year international collaboration that will develop such multiscale predictive models for the musculoskeletal apparatus. The aim of this paper is to update the reader on the recent progress of this initiative. In particular, we shall report on the development of the Living Human Digital Library, an infrastructure to support the collaboration of multiple research institutions on the LHP. We shall also describe how these complex models can be used to generate subject-specific predictions, which is highly relevant for clinical practice.

Key words: virtual physiological human, multiscale modelling, musculoskeletal system.

1 INTRODUCTION

1.1 Why multiscale?

A group of genetically identical inbred mice are housed under the same conditions; half of them are subjected to mild (0.3 g) whole-body mechanical vibrations for a few minutes per day. After a few days, the transcriptional activity of hundreds of genes related to the musculoskeletal apparatus becomes significantly different between the two groups of mice[1]. To explain this observation, we have to find the systemic relationship that links the accelerations experienced by the mouse's body with the transcriptional activity of its genes. But this is possible only if we can model the dynamics of the whole body, the biomechanical process that transfers the signals down to the tissue, the mechano-biological transduction through which these signals modulate the behaviour of the cells, and the process within the cell that alters the transcriptional activity of the genes. In other words, we need to develop an understanding of the physical processes that are occurring at the various dimensional scales, and how they interact with each other.

2.2 The Physiome

The concept of a "Physiome Project" was presented in a report from the Commission on Bioengineering in Physiology to the Council of the International Union of Physiological Sciences (IUPS) at the 32nd World Congress in Glasgow in 1993. The term "physiome" comes from "physio-" (life) and "-ome" (as a whole), and is intended to provide a "quantitative description of physiological dynamics and functional behaviour of the intact organism". It first appears in the Medline index in 1997, in a paper of Jim Bassingthwaighte on the Cardiome project[2]. While the Physiome project has undoubtedly been successful in pulling together some of the best research groups in the world around the specific problem of cardiac modelling, the initiative has so far failed to take on the global dimension that its proponents have expected. Nevertheless, considerable attention has been generated around this topic recently, particularly in Europe.

3.3 The STEP action

This attention recently condensed into the STEP coordination action, which started at the beginning of 2006 under the leadership of some of the most important institutions in Europe in the area of Physiome research. STEP strongly promoted inclusiveness and adopted a very broad consensus process directly involving hundreds of experts world wide and presenting information to many thousands of potentially interested professionals. The expert panels included not only top-level researchers, but also representative from clinical, industrial and societal organisations, so as to reflect the perspective of all relevant stakeholders in the consensus process; they also included many participants from outside Europe in order that the developments were informed by work taking place in other parts of the world. The main products of this consensus process are the EuroPhysiome initiative, and a research Roadmap for the European Research System that suggests an approach towards a true Grand Challenge – the *Virtual Physiological Human*.

4.4 The VPH and the VPH road map

The current approaches to investigating the human body are analagous to attempting to discover the picture portrayed by a complex jigsaw puzzle made of a trillion pieces by looking only at a single piece or, maybe, a few closely interconnected pieces; it should, therefore, come as no surprise that we are not finding it easy.

In order to continue the scientific exploration of the human body that has already so dramatically improved the length and quality of the life for a major part of the humanity, it is necessary to complement this traditional *reductionistic* approach with what the STEP experts call an *integrationistic* approach, that is, the combination of observations, theories and predictions across the temporal and dimensional scales, across the scientific disciplines, and across the anatomical sub-systems. This integrative approach requires a radical transformation of the way in which biomedical research is conducted. It is necessary to create a framework within which observations and measurements from a variety of sources can be collected, shared and combined in many different ways.

This framework should allow experts from a variety of disciplines to work collaboratively to analyse these observations and develop systemic hypotheses. It should also make it possible to combine predictive models defined at different scales or with different methods or with different levels of detail, in order to make the hypotheses concrete, and to allow their validity to be tested against existing results. The scope of the EuroPhysiome Initiative is to promote the development of the Virtual Physiological Human (VPH) “a methodological and technological framework that will enable investigations of the human body to consider it simply as a single (though hugely complex) system”.

The VPH Research Road Map[3] describes the research activities that the European research system will have to pursue in the next ten years in order to realise and deploy the VPH. While the VPH will have a sizeable impact on all branches of biomedical research and clinical medicine, two domains are clearly primary targets: cardiovascular and musculoskeletal. The second domain is the target of the *Living Human Project* (LHP).

5.5 The Living Human Project

The Living Human Project will develop a worldwide, distributed repository of anatomo-functional data and of simulation algorithms, fully integrated into a seamless simulation environment and directly accessible by any researcher in the world. This infrastructure will be used to create the physiome of the human musculoskeletal system.

In marked contrast to other initiatives, the LHP approach is top-down – starting with whole body modelling and then drilling down to the cellular level. It must be noted that, although the LHP was conceived long before the VPH Roadmap was published, its aims, even in the early days, were perfectly aligned with what has come out from the Roadmap. In particular, the LHP has long recognised that the creation of an infrastructure that makes the collaborative development of a complete multiscale model of the musculoskeletal system possible is a major priority; indeed, the first major grant received by the LHP is directed towards the development of such an infrastructure, called the *Living Human Digital Library* (LHDL). The LHDL project began in February 2006 and will for last three years. In the following section we shall describe how, in the current analysis, the final system will work from a user perspective, and from a general architecture standpoint.

2 THE LIVING HUMAN DIGITAL LIBRARY: USER EXPERIENCE

The LHDL services are available within a virtual community called Biomed Town[4], which is dedicated to students and professionals involved in biomedical research. Membership of Biomed Town is free, but during the development phase, access to LHDL services will be limited to selected beta users; if you are interested in becoming a beta user, please contact the first author. All services provided by the LHDL will be accessible through a web portal[5](Figure 1).

The fundamental design of the LHDL revolves around the integration of a Grid application mostly based on web services and other web technologies, and of an application running locally on the client PC, called *LhpBuilder*. This software is being developed using a framework called the *Multimod Application Framework* (MAF), which is designed to support the rapid development of biomedical software. For further information on MAF and related concepts, please refer to the recent descriptive paper [6]. All of the data generation and most of the data interaction take place inside LhpBuilder, while the data management, the data storage and the execution of any computer-intensive tasks are performed within the Grid infrastructure.

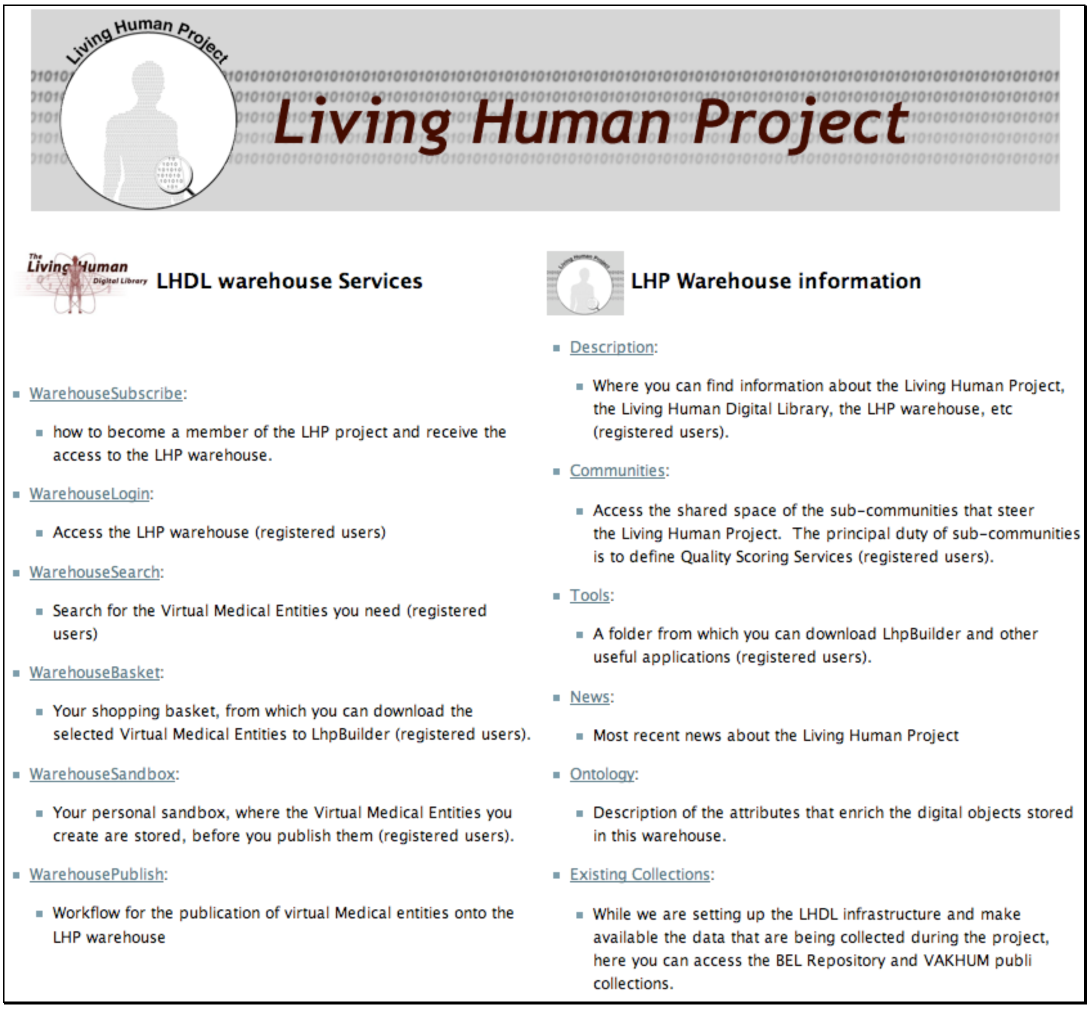

Figure 1. The LHDL web portal, through which the user can access all available services

1.1 Creating new resources

The user can create new resources using LhpBuilder and the MAF. Inside any MAF application, including LhpBuilder, resources are provided in the form of data objects called *Virtual Medical Entities* (VMEs)[6]. As we shall see, they should wrap both data and services, which are hereafter referred to, generically, as resources. resources that are collected for use in a particular study can be organised into what is called a VmeTree. From inside LhpBuilder, we can create a new VmeTree, which is initially empty; this can then be populated with instances of various VME types, using several different methods.

The first method is to create a data VME by importing a digital dataset. When a dataset is imported, a new VME of the appropriate type is created inside the VmeTree. LhpBuilder has importers for most of the formats commonly used in biomedical modelling, including DICOM for medical imaging data, STL and VRML for geometric data, C3D for motion capture data, etc. The MAF makes it relatively easy to add a new importer if required; thus, even if your datasets are written in an arcane format, with a little C++ knowledge it is possible to add a new importer for your datasets to the MAF (and thus to the LhpBuilder).

MAF is an Open Source project; every user can thus find instructions on how to develop additions and how to submit them for inclusion in the next release. MAF is distributed under an BSD-like licence model, and each contributor owns the intellectual property rights of the code included; these two conditions ensure that MAF is, and will remain, open source, and that anyone can use it to develop commercial applications without paying royalties, etc. MAF makes available a complex set of data structures that should accommodate most biomedical data types (Figure 2).

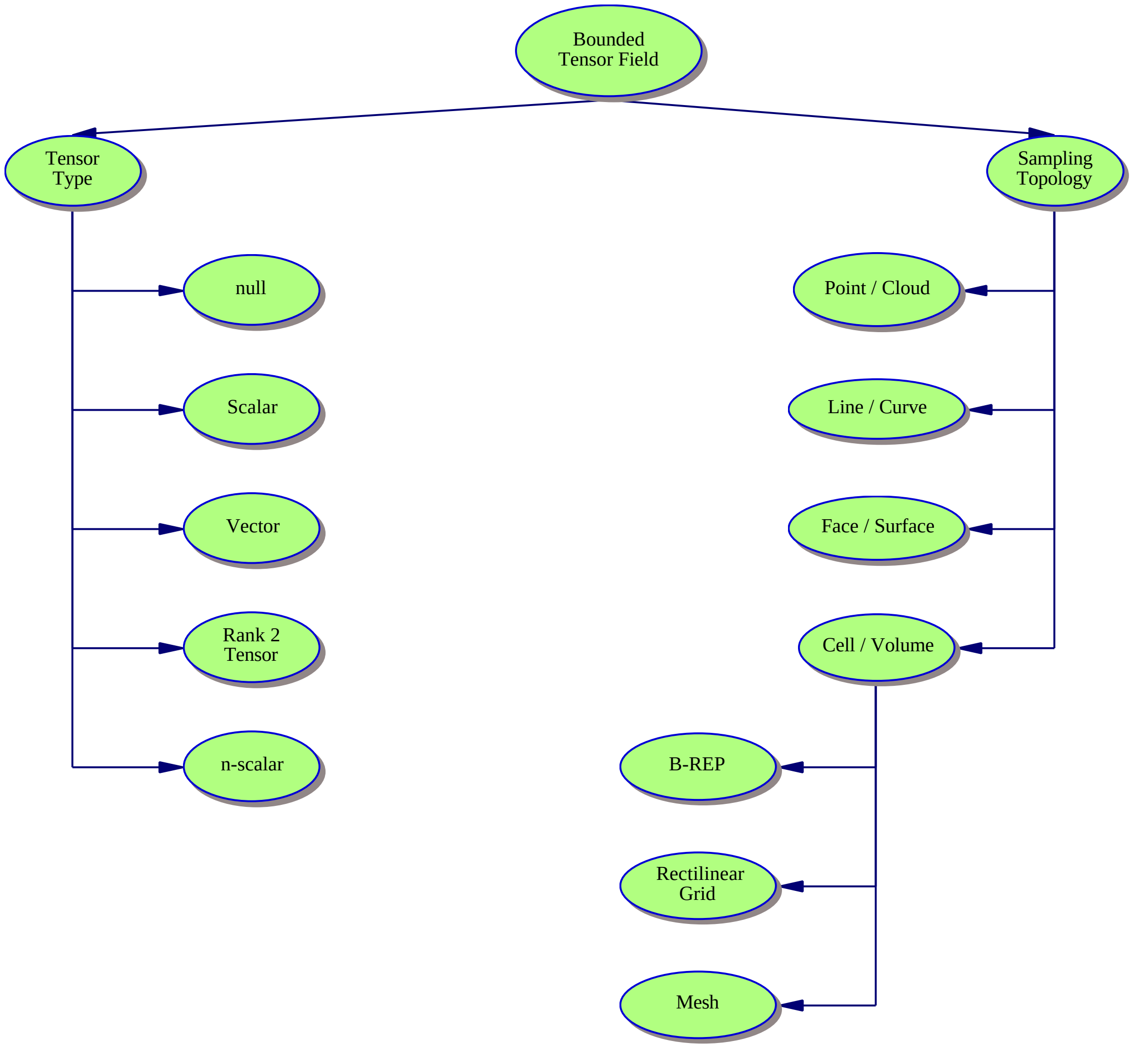

Figure 2. Data types that are supported by the LhpBuilder application. All data objects can be defined as time varying in the data and in the pose. The pose of a VME is defined with respect to the parent VME in the VmeTree

A second method for creating VMEs is to use an Operation – Operations are procedures available in side LhpBuilder that allow the user to create or modify VMEs.. Thus, one can either use a *Create* Operation directly or make a copy of an existing VME and then alter it with a *Modify* Operation.

A third mechanism is the use of a *Derived VME*. This special type of VME contains a dataset that is not static, but rather the result of a procedural calculation defined inside a *data pipeline*; the data pipeline can be executed locally or remotely, which provides a basic mechanism for remote execution. The dataset inside a derived VME can be computed by using inputs provided by the user and/or by reading other VMEs that are already available in the VmeTree. With this latter option, one can create a *Transformation VME*, which generates a new VME that is the transformation of an existing VME according to certain procedures.

Irrespective of which method is employed, the new VME is initially created as child of the *Root VME*, that is, the base node of the VmeTree; if desired, the user can subsequently move the VME to any location in the tree, creating a complex structure – a binary tree in which each VME can be parent of a VME and child of another VME. The VME also contains a pose matrix that defines its position and orientation in space with respect to the parent VME. When a new VME is created, its pose matrix is set to the identity, but with certain Operations, it is possible to move and rotate the VMEs, by changing their pose matrix.

2.2 Submitting new resources to the repository

Once new VMEs have been created inside the LhpBuilder, the user can, with a single command, upload it into the LHDL. The new data resource is created inside a logical space that is the user sandbox. Until the resource is there, only the user can see it. To make it public, the user must enrich it with all the necessary meta-information, and publish it.

When the user uploads a data object into the LHDL, LhpBuilder generates an amount of meta-information relating to the object itself. Examples are the creation date, the VME type, the source of the object (the parent object if created by editing, the name and type of the data file if imported), the size, relevant object metrics (polygon counts, voxel spacing) and, in general, all of those attributes that the program can generate automatically.

In a number of situations, the real information is not in the binary object, but in the relationships (spatial, temporal, logical) that the various data objects have to each other. The only way to preserve this information when posting the data to the LHDL is to upload, not the single data object, but the entire VME tree. However, in order to avoid replication of storage (typically any VME tree contains some new data objects together with many others that are already available in the repository) and potential infringements of ownership and usage agreements, the system should be able to retain a unifying resource ID, which is recognised for pre-existing data resources.

The VME tree, that is, the spatial/temporal/logical relationships defined among the various data resources, would become a data resource itself. When you search for a data resource, you may request that the system lists all the VME tree resources to which this particular data resource belongs.

Whenever the user publishes a resource, the system attaches to that resource an instance of the current meta-information set, which the owner has to populate. Some metadata are mandatory – if they are not populated, the resource cannot be published and it will remain in the sandbox; others are optional, but the completeness of the meta-information is a primary quality indicator for each resource.

When the metadata are modified, added or extended, the system should automatically create a list of all the resources that are affected by this change, and inform the owners of these resources that they need to revise the meta-information. In this way, we shall force the community to take responsibility for ensuring that the collection is always well aligned with the current domain ontology.

Once a data resource has all of its metadata populated, it is ready for publication. For this, the user must choose from a set of pre-defined sharing policies by which the price is established, in units, for each sub-community and for each access type. The publication is an irreversible action; once published a resource cannot be modified, so as to ensure the uniqueness of that resource and its association with the unique resource ID.

If changes to a resource are required, the user can publish a replacement resource, and add to the older one a link which states that it is obsolete and has been replaced by the newer one, as done with printed publications.

3.3 Accessing, searching and using available resources

The first step that any user should take is to become a member of the LHP virtual laboratory. Joining the virtual lab is a relatively simple operation, which involves only filling up some web forms with personal and institutional details. However, this leaves the user as little more that a visitor in the virtual lab.

As we shall see, inside the virtual lab, the user will be able to perform activities of increasing responsibility only when he/she is explicitly authorised to do so by the community and/or by "the Operator", the entity that will operate the LHDL after the end of this project. Hereafter, we shall assume that the user has all the permissions necessary to perform the activities described. It will be the responsibility of the community and the Operator to define the protocols that govern how and when a user is upgraded to particular levels of responsibility within the community.

After logging on, the user will be exposed to a search environment that should capitalise on the domain ontology and on the enrichment of meta-data for each resource to provide a very powerful search and discovery environment. The user searches the digital library for the various VMEs needed, and adds them to a shopping basket, following a metaphor made popular by e-commerce web sites. When all the necessary resources have been gathered, the user will download the contents of the basket as a single compressed file that will be opened by LhpBuilder. If some VMEs added to the basket originally belonged to the same VmeTree, the construction of the new VmeTree will attempt to re-establish as many of the hierarchical relationships between the VMEs as possible; thus, if a surface VME was created as child of the Volume VME it segmented, when we add both VMEs to the basket, they will again be placed in a parent-child relationship in the new VmeTree.

4.4 Using data resources within LhpBuilder

Once the selected data are open inside LhpBuilder, the user will be able to visualise them, modify them, create new data through operations, etc. LhpBuilder will execute as a local program on the client machine and should be able to operate as a stand-alone program, even if not connected to the LHDL.

In this sense, LhpBuilder will replace Data Manager, an older MAF program that was reported previously in studies by some of the authors[7, 8]. The most common use of LhpBuilder will almost certainly be the interactive visualisation. All LHP data resources will have visualisation pipelines to enable them to be represented visually in the most appropriate way. It will also be possible to perform a number of operations that do not require the original dataset to be edited, such as measurements, virtual palpation, classification/segmentation, registration/synchronisation of another dataset on to the original one, etc.

Each data resource should come with a permission flag, which can be modified only by the owner, that defines the level of access that the user can have to that resource: visualise only, edit, and export. *Visualise* will allow the user to see the data in various ways, to take measurements, and to perform operations that do not require a copy of the dataset to be made. *Edit* will allow the user to create new datasets by editing the original one, while *Export* will enable the user to export the dataset into an external file, written in a neutral format that allows the file to be read into other programs. In order to prevent tampering, it will probably be necessary to transmit all LHP data in an encrypted format.

Assuming that the data resource comes with the necessary permissions, the user will be able to edit it within LhpBuilder by invoking suitable operations. In general, any edit operation will firstly create a copy of the dataset and then modify it, so as to preserve the original data quantum. *Edit* operations include direct data editing, spatial and temporal transformations, etc. The new dataset will be stored in the library with the clear indication that it is derivate (*Synthetic*, as opposed to *Natural*, to use the jargon employed in Data Manager), the identity of its source dataset and, possibly, which operations were applied to generate it.

Again, if the data resource comes with the necessary permissions, the user will be able to export the dataset into an external file written in neutral formats that other programs can read. Typical targets are image processing programs, simulation codes, etc.

Another possibility will be for the user to process one of the available resources with a *service*. This part of the LHDL system will be developed in the final phase of the project and is currently in the conceptual design phase.

5.5 Selling and buying LHP units

One of the major problems associated with all initiatives such as the LHDL is long-term sustainability, that is, how to continue the operation when the initial funding runs out. In LHDL, we plan to carry out some experimentation on an original model that we think could provide such sustainability without being too intrusive in the users' experience.

In the early stages of the project, properly authenticated users will be given a "digital wallet" containing a certain amount of credit to support purchasing or bartering within the community. The minimum quantum of credit will be called a unit. Each resource available (and possibly each use of that resource) will have a price in units, and the user will be allowed to use or acquire it only if his/her digital wallet contains sufficient credit. The units are transferred to the digital wallet of the owner of the resources, apart for a small commission collected by the operator for the costs of the transaction.

During the project, the Operator will give new users the units that it collects, since the costs are sustained by the project budget. At the end of the project, this share will be cached and will then constitute the profit that will ensure that the operator can stay in business indeterminately. Of course, in a second stage, users will also be allowed to buy units, not just barter them, which will make it possible for the industrial users to enter the game.

6.6 Brief description of the technological infrastructure

The LHDL project is still in its early stages, so the precise details of the architecture that will ultimately be deployed are still being developed. However, a preliminary description will give a good idea of the level of complexity that is required to provide the users with the type of services described above. A schematic representation of the LHDL architecture (Figure 3) includes the following blocks.

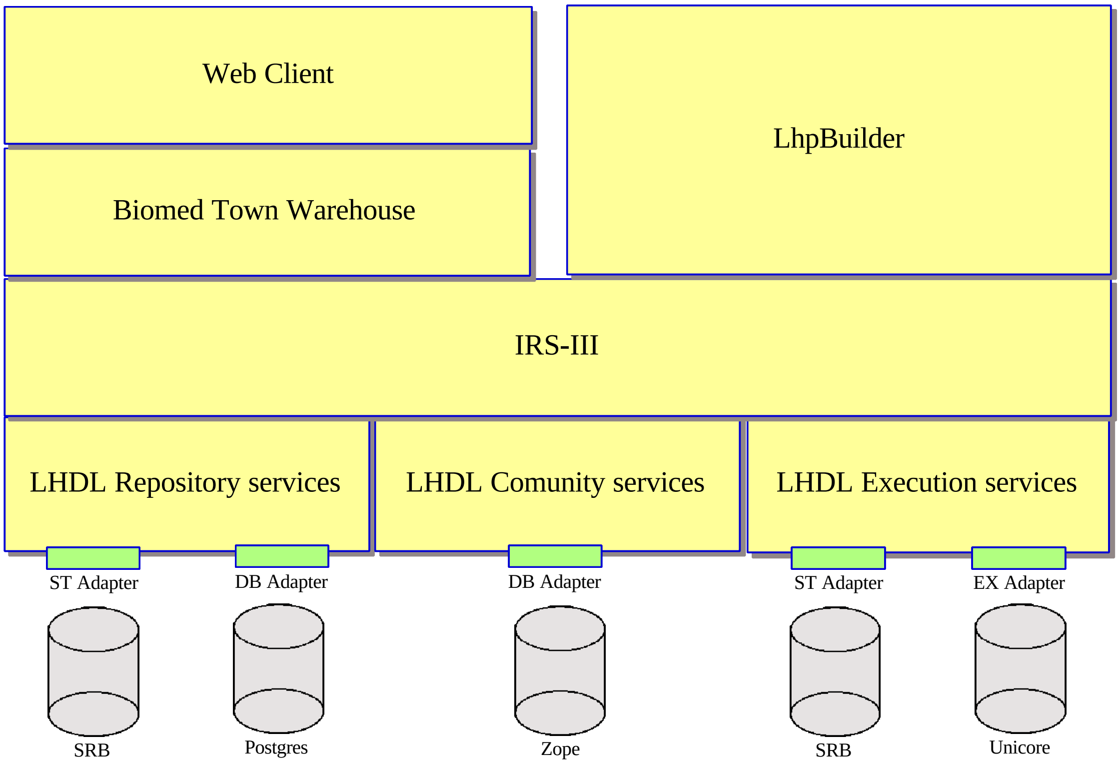

Figure 3. Schematic representation of the LHDL architecture and of its main components, including some preliminary indication of the technologies that we plan to use for the deployment

As we have already indicated in the description of the user experience, we can access LHDL via two interfaces: the MAF client application (LhpBuilder) and a web client. The browser will access Biomed Town and, through it, will access IRS-III[9], a semantic broker that mediates among user "goals" and exposed web service. There are three types of backend web services: *community services* that manage the user profiles, the textual databases, etc.; *repository services* that manage the storage and retrieval of resources, and *execution services* that manage the remote execution of data pipelines.

3 A SEMANTIC LHDL

1.1 Toward the VPH

The basic architecture of the LHDL is relatively simple and can be probably managed with conventional approaches. However, if we project the LHDL in the future development of the VPH, we shall see that the scenario becomes much more complex.

The development of the VPH will follow complex paths; in some cases the research groups will form around specific pathologies, in other cases around subsystems. Each group will express a certain community, and will probably deploy its own infrastructure, made of storage, execution, community and semantic services.

In principle a central organisation (e.g. the EuroPhysiome initiative) could manage a VPH portal, which would provide to all users a single entry point to this cloud of VPH sub-communities. However, the various resources made available by each project/community would remain separate. So for example, if I am looking for an ECG recording, this may be available either at the Cardiome repository or at the @neurIST repository. With a VPH architecture federated only at the VPH portal level, I would have to join each sub-community, make a search and see if I can find what I am looking for. As the number of VPH sub-communities will increase and as the number and the complexity of service they provide will also increase, this scenario becomes impossible to manage. As a result, each VPH researcher will tend to use only the services provided by its community, missing part of the potential of the VPH.

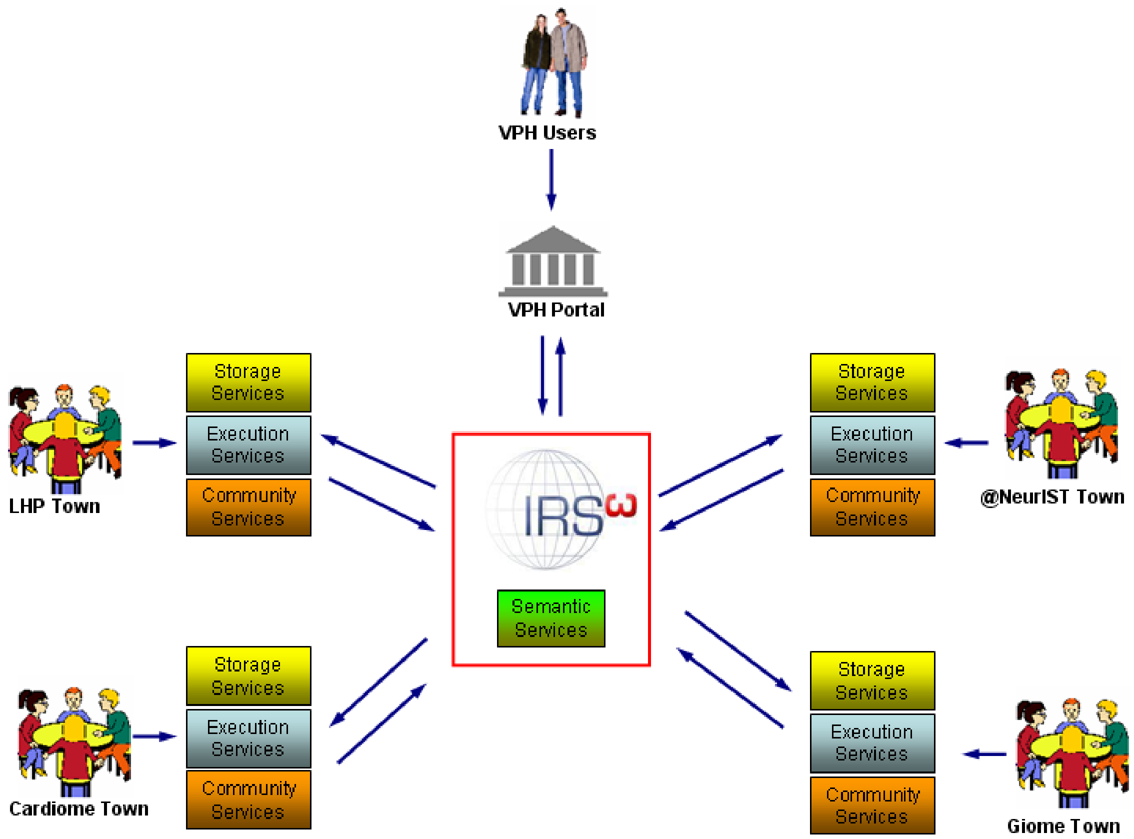

Figure 4. Scenario: IRS-III as semantic broker among a VPH Portal and several communities

2.2 Federating VPH sub-communities

There are multiple levels at which the VPH communities and the services they develop can be federated.

A *centralised Portal* gives a first simple level of federation. An umbrella organisation, e.g. an evolution of the Europhysiome group emerged by the STEP action, could operate a portal that acts as a directory service for all VPH resources available on the Internet. Such portal should provide an easy-to-use method to register a VPH resource (communities, data and models warehouses, software repositories, ontologies, etc.) so that the users can get a detailed listing of everything is available “out there”.

At the other extreme of the stack, storage and computing resources used by VPH services could be federated using *Grid technology*. Grid computing is evolving quickly and its definition varies considerably depending on the author and the application context[10]. A key point of Grid technology is the need to share ICT resources between organisations; this definition implies that these resources are so precious that they cannot be replicated at all sites. Under this perspective, we fell that Grid technology is relevant only when High Performance Computing Networking Storage and Analysis (HPC) resources are involved. If this is true Grid technology is relevant to the VPH as far as HPC resources are involved. However, since in HPC resources the need for efficient usage impose a strong coupling between hardware and software, and complex management of these resources, from a VPH perspective HPC resources, although deployed on Grid infrastructures, should be exposed as highly-abstracted services.

One possible approach to build complex ICT architectures from federated services is the so-called *Service-Oriented Architecture* (SOA). SOA describes a software architecture that defines the use of loosely coupled software services to support the requirements of business processes and software users. Resources in an SOA environment are made available as independent services that can be accessed without knowledge of their underlying platform implementation[11]. SOA can be used to build Grid-based applications that are more transparent and supported on a wider range of platforms and environments. Service-oriented architectures are not tied to a specific technology; however, the most common way to deploy SOA are the *Web Services*.

A web service is a software system designed to support interoperable Machine-to-Machine interaction over a network. Although the W3C Web service definition[12] encompasses many different systems, in common usage the term refers to those services that use the SOAP messaging protocol to exchange XML messages transported by the HTTP transport protocol and described by the Web Services Description Language (WSDL).

3.3 Semantic Web Services in VPH

Using Web services, data and functionalities can be shared easily over the Internet. Services can be supplied as autonomous platform-independent computational elements with a relatively good application performance. However, current syntactic based standards do not describe the capability of a service and cannot be understood by software programs. A human developer is required to interpret the meaning of the Web service inputs, outputs and applicability constraints, as well as the context in which the service can actually be used. Moreover, Web services lack flexibility; for instance, if a new Web service is deployed to VPH community, the application developers will have to create and re-align several descriptions in order to maintain the integrated application. On the other hand, a semantic based approach will normally capture the knowledge associated with the background context together to the requested and provided capabilities, and hence support automatic reasoning and reuse. In this way, service invocation, discovery, composition, and mediation can be automated by adopting the best available solutions for a specific request increasing the flexibility, scalability, and maintainability of an application.

The Semantic Web[13]is an extension of the current Web where documents incorporate machine processable meaning. The overall semantic Web vision is that one day it will be possible to delegate non-trivial tasks, such as finding all pathologies related with a number of symptoms, to computer based agents able to locate and reason with relevant heterogeneous online resources. One of the key building blocks for the semantic Web is the notion of an ontology[14]. An ontology is an explicit formal shared conceptualization of a domain of discourse. More specifically, an ontology captures the main concepts and relations that a community shares over a particular domain. Within the context of the semantic Web ontologies facilitate interoperability as the underlying meaning of terms within a Web document can be made explicit for computer based agents to support processing.

Semantic Web services (SWS) research aims to automate the development of Web service based applications through semantic Web technology. By providing formal descriptions with well-defined semantics we facilitate the machine interpretation of Web service descriptions.

“In a Service-Oriented Architecture (SOA), semantic interoperability ensures that service consumers and providers exchange data in a consistent, flexible way that fulfils non-functional requirements (NFRs) such as performance and scalability, regardless of the diverse information involved”.

4.4 Our Approach: semantic interoperability

We strongly believe that SWS technologies make interoperability a reality in a huge context as the biomedical one. For this purpose, we are exploring a new approach within LHDL project, which takes advantages of SWS potentialities.

The idea is adopting, *Internet Reasoning Service* (IRS-III) [9], as semantic broker, for alleviating most of problems that usually make difficult a full interoperability between heterogeneous resources may be deployed in VPH, such as finding, composing, and resolving mismatches between Web service components.

IRS-III is a platform, designed and implemented by Knowledge Media Institute of the Open University (UK), for developing and executing semantic Web services. In particular, IRS-III has incorporated and extended the Web services Modelling Ontology (WSMO)[15] as the epistemological framework. In the following sections we will describe more details of WSMO and IRS-III.

WSMO

The Web Service Modelling Ontology (WSMO) is a formal ontology for describing the various aspects of services in order to enable the automation of Web service discovery, composition, mediation and invocation. The meta-model of WSMO defines four top-level elements:

Ontologies provide the foundation for describing domains semantically. They are used by the three other WSMO elements;

Goals define the tasks that a service requester expects a Web service to fulfil. In this sense they express the requester's intent;

Web Service descriptions represent the functional behaviour of an existing deployed web service. The description also outlines how web services communicate (choreography) and how they are composed (orchestration);

Mediators handle data and process interoperability issues that arise when handling heterogeneous systems.

One of the main characterizing features of WSMO is the linking of ontologies, goals and web services by mediators, which map between different ontological concepts within specific WSMO entity descriptions. In order to facilitate appropriate mapping mechanisms, four classes of mediators are considered within WSMO. For example, an OO-mediator may specify an ontology mapping between two ontologies whereas a GG-mediator may specify a process or data transformation between two goals.

IRS-III

IRS-III is a platform and a broker for developing and executing semantic Web services. By definition, a broker is an entity that mediates between two parties and IRS-III mediates between a service requester and one or more service providers. To achieve this, IRS-III adopts a semantic Web based approach and is thus founded on ontological descriptions. In particular, we have incorporated and extended the Web Services Modelling Ontology as the core IRS-III epistemological framework.

A core design principle for IRS-III is to support capability-based invocation. A client sends a request which captures a desired outcome or goal and, using a set of semantic Web service descriptions, IRS-III will: a) discover potentially relevant Web services; b) select the set of Web services which best fit the incoming request; c) mediate any mismatches at the data, ontological or business process level; and d) invoke the selected Web services whilst adhering to any data, control flow and Web service invocation constraints.

Additionally, IRS-III supports the SWS developer at design time by providing a set of tools for defining, editing and managing a library of semantic descriptions and also for grounding the descriptions to either a standard Web service with a WSDL description, a Web application available through an HTTP GET request, or code written in a standard programming language (currently Java and Common Lisp).

The main components of the IRS-III architecture are the IRS-III Server, the IRS-III Publisher and the IRS-III Client, which communicate through the SOAP protocol.

IRS-III was designed for ease of use, in fact a key feature of IRS-III is that Web service invocation is capability driven. The IRS-III Client supports this by providing a goal-centric invocation mechanism. An IRS-III user simply asks for a goal to be solved and the IRS-III broker locates an appropriate Web service semantic description and then invokes the underlying deployed Web service.

5.5 Semantic interoperability in LHDL

A core design principle for IRS-III is to support capability-based invocation. A VPH user performs a request by the VPH portal; the portal sends the request which captures a desired outcome or goal and, using a set of semantic Web service descriptions, IRS-III will: a) discover potentially relevant Web services in any sub-community; b) select the set of Web services which best fit the incoming request; c) mediate any mismatches at the data, ontological or business process level; and d) invoke the selected Web services whilst adhering to any data, control flow and Web service invocation constraints.

In our approach, the ontologies define the common terminology, and the web services represent the services that each VPH community will expose. Te execution sequence of a complex semantic Web services is not hard-coded, but it is dynamically created using goal-based discovery and invocation: several Web services may be associated with a goal, and only the most applicable will be discovered and invoked at runtime (late binding); if a new service will be available, the developers simply need to describe and then link it to an existing goal; if a service is altered, only the specific semantic description will be affected, and not the whole business process

In principle a centralised authority could take the responsibility to define via consensus processes the common goals, the common ontologies and the web services descriptions. However, the use of the Mediators makes possible to accommodate into a coherent representation goals, ontologies and web services that could be developed independently by each sub-community.

In general there will be mismatches between the goal requests and available Web services and between the goals themselves. The IRS-III mediation handler components are responsible for resolving the conceptual mismatches which may occur by reasoning over the given goal, Web service and mediator descriptions. Thus, mediators can mediate at data level (between ontologies), among goal representations, and at process level.

Once the number of VPH sub-communities will be large enough, and the services they will expose will be sufficiently large, most of the development in the VPH will be done at the semantic level, by cleverly create new composite services orchestrating the existing ones, and by defining new original user goals that make innovative uses of existing resources.

While this scenario might appear too far to be relevant here, it is already clear that in order to become effective the VPH will have to huge; thus, it is mandatory that any architectural design consider the scalability and the increase of complexity to the highest possible levels from the very beginning. The alternative is to develop infrastructures that will inexorably fail as soon as the VPH will reach its critical mass.

6.6 Communication between IRS-III and Web services deployed by VPH sub-communities

As mentioned earlier, the IRS-III components are specified as combination of an ontological meta-layer and WSMO definitions. During communication with a Web service, the ontological level descriptions need to be mapped to the XML based representations used by the specific Web service invoked. IRS-III provides two mechanisms that map: a) from the ontological level to XML (lower); and, b) from XML to the ontological level (lift).

Lift. Lifts an XML string to an ontological construct, represented in OCML[16], the modelling language used within IRS-III. A generic version of this relation is defined within the IRS ontology. SWS developers are free to overwrite this relation inline with the relationship between the results of Web service calls and the ontologies used. The lift primitive has the following input parameters: class-name, web-service-class, xml-string and produces an instance of class-name as output. The semantic developer can thus customize how XML is parsed according to the classes within the underlying ontology and the particular Web services selected. In order to cope with input in XML format the lift primitive utilizes an inbuilt SAX based XML parser.

Lower. Lowers the ontological construct to XML. The input parameters to lower are: instance-name and a Web service class. The output is xml-string. As for the lift primitive, the XML generated can be customized according to classes within the ontology and the Web service class. For example, the XML generated for instances of a person class may include a full name for one Web service and only a family name for another.

7.7 LHDL User Goals

The WSMO representation of the LHDL is an on-going process. However, to provide an example of the work being done, we shall provide some preliminary results.

The analysis of the users needs produced the description of 26 independent user goals. We have categorised them as Mandatory, Necessary and Useful:

Mandatory

1. Register with the LHDL service
2. Access the LHDL service
3. Search and retrieve data resources
4. Use data resources within LhpBuilder
5. Import data into LhpBuilder
6. Submit to the repository new data resources
7. Download Data Resources
8. Upload Data Resources

Necessary

1. Create new data resources by editing an existing one
2. Export data resources to external programs
3. Enrich the data resources by adding meta-information
4. Publish new data resources
5. Search and retrieve service resources
6. Create new processing resource for the LhpBuilder

Useful

1. Create new data resources by processing existing ones with web services
2. Transform an LhpBuilder processing resource into a LHP web service

3. Sell and buy LHP units
4. Report malfunctioning of the digital library
5. Find help on how to use the LHDL
6. Policy Manipulation
7. Resource Monitoring
8. Restore Data Backup
9. User Profile Management
10. Data Access Auditing
11. Data Access Mining

Each goal specifies the objectives that a client may have when consulting a Web Service, describing aspects related to user desires with respect to the requested functionality and behaviour. Ontologies are used as the semantically defined terminology for goal specification. Goals model the user view in the Web Service usage process and therefore are a separate top-level entity.

Inputs and **Output** are mandatory parameters in our goal definition, they specify the inputs (name and type) necessary to achieve the goal, and the output is expected.

Precondition, **Assumption**, **Postcondition**, and **Effect** are optional axioms. Preconditions specify the information space of the Web service before its execution. Assumptions describe the state of the world before the execution of the Web service. Postconditions describe the information space of the Web service after the execution of the Web service. Effects describe the state of the world after the execution of the Web service.

As an example with report here the semantic analysis of the user goal, *Search and retrieve data resources.*

The goal “Search and retrieve data resources” represents locating data resources, called VME, that the user is looking for. The user can perform a search by using one or a combination of the following criteria:

Search by donor attributes

Search by data type

Search by acquisition parameters

Search by anatomical location/relation

Search by creation attributes

Every type of search is carried out by a proper web service, which defines its capability. Essentially, the search is expressed by a list of criteria, declared in the web service effects. In other words, the goal “Search and retrieve data resources” is semantically defined as one only goal; different web services will solve all types of searches.

We model “search and retrieve data resources” by a class search-goal, which has inputs and output according to its requirements, and respectively performed by the classes search-input and search-output (Figure 5).

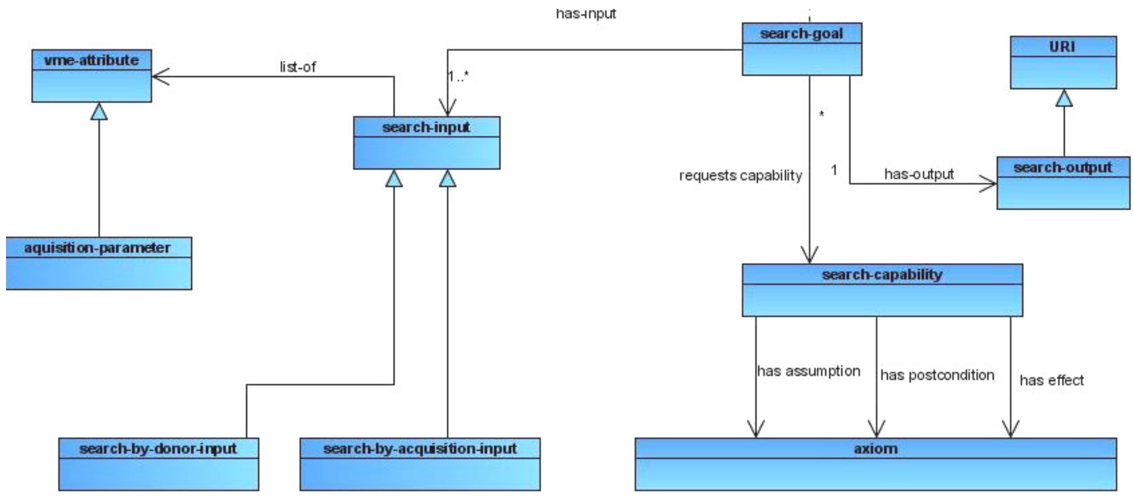

Figure 5. Search and retrieve data resources goal

The class search-input stores all possible inputs have to be instantiated in order to achieve the goal. The search-goal inputs are represented by lists of VME entities (class vme-entity) Listing 1 opportunely defines the taxonomy (first level) of the class vme-attribute.

```
Class vme-attribute
subclass-of (subject-attribute, vme-attribute)
subclass-of (quality-attribute, vme-attribute)
subclass-of (domain-attribute, vme-attribute)
subclass-of (data-attribute, vme-attribute)
subclass-of (file-attribute, vme- attribute)
subclass-of (tree-attribute, vme-attribute)
subclass-of (policy-attribute, vme-attribute)
```

Listing 1 Direct subclasses of vme-attribute class

The output the user expects to have back is the URI with all information he requested. The relation request-capability is inherited from WSMO model; within the capability precondition, postcondition, effects assumptions are defined by axioms. In this case, the search goal does not expect any precondition. In order to represent the effect, we can perform in the ontology the class visited-URI sub class of the URI. Visited-URI stores the URIs already visited, then all search results. The following logical expression performs the search goal effect:

If instance-of(X,URI) Ù instance-of(S, search-goal) Ù has-output(S, X)

Þ

instance-of(X,visited-URI).

The postconditions are different in any type of search, because different criteria have to by applied. Proper web services will satisfy the postcondition. We are going to provide more details later in this section, by analyzing all type of search.

For every different kind of search, we need to instantiate different lists of inputs. For this purpose, we introduce the following sub-classes of search-input: search-by-donor-input, search-by-data-type-input, search-by-acquisition-input, search-by-anatomical-location-input, search-by-creation-input (Figure 5 shows only the search-by-donor-input and search-by-acquisition-input classes).

A kind of search we are implementing is a *Search by data type*.

The taxonomy of *vme-attribute* (see Listing 1) can be further extended: *data-type*, *creation* and *instrumentation* are subclasses of *data-attribute*. In turn, *image, signal* and *measurement* are *data-type*'s subclasses.

The context we are describing is following: the user would like to find all VME, type Volume, medical imaging modality CT lying in the region near to pelvis. She submits this query by fulfilling the proper fields in a Web form available on VPH portal. The Web service, that a VPH sub-community deploys, is not able to achieve the user's goal, because it does not "understand" the concept "near to". In turn, another Web services deployed by another sub-community is able to associate a body region to a list of bones near to that region.

Within IRS-III are semantic descriptions of both Web services. Thus IRS-III, by composing two services, and in case, by solving any mismatch, is able satisfy the user goal. The all process is detailed in the following section.

8.8 Search service analysis

1) Within VPH resides a form for allowing the user express its goal, by the suitable fields. The goal is "search of the data VME", particularly, in this prototype, it can be modelled by the query: "find all VME of type Volume, medical imaging modality CT of the region near to pelvis". The user fill all required fields and click the submit button.
2) VPH portal calls the IRS-III by a HTTP GET request. This request includes IRS-III location, the function for calling the goal (achieve-goal), the goal name, and the inputs specified by the user.
3) Two Web services, "search Web service" and "location-bones-mapping Web service" are deployed by different VPH sub-communities. The first one is able to solve queries like "find all VME of type Volume, medical imaging modality CT of the bones femur or sacrum". The "location-bones-mapping", receiving in inputs an anatomical location; it gives back in output a list of bones near to the given body location.
4) Within IRS-III there are the semantic descriptions of both search Web service and location-bones-mapping Web service. The two services are connected by wg mediators with semantic representations of goals. The composition of these two goals represents the user goal, also semantically represented in IRS-III.
5) IRS-III sends a request to location-bones-mapping Web service in XML format, according the protocol used by Web service (SOAP, XML-RPC). The request that IRS-III sends includes a pair (attribute, value), i.e. (anatomical location, value chosen by the user). IRS-III can submit "syntactic" queries thank to lowering process described before.
6) The location-bones-mapping Web service processes the query, like "find all bones near to region pelvis" and to obtains a list of bones near to the given anatomical location.
7) The result is returned to IRS-III as XML format, that provides the bones list. IRS-III lifts at an ontological level the XML data received.
8) IRS-III sends another request to Search Web service that resides in another sub-community The request includes, as inputs, pairs (attribute, value) like: (VME of type, volume), (medical imaging modality, CT), (bones, (femur, sacrum)). The third input comes from the location-bones-mapping service output.
9) The Search Web service solves a query such as "find all VME of type Volume, medical imaging modality CT of the region = (femur OR sacrum)"
10) The result of the search is returned to IRS-III in a XML format.
11) The output IRS-III receives from the repository Web Service is returned to VPH

portal in XML or HTML format.
12) VPT renders the result in a HTML form.
13) The browser displays to the user the result of the semantic search; from this web page the user can selected which VME she likes to download and transfer them into the basket

9.9 Conclusions

From this example it is possible to understand how general is this approach and how useful it can be to manage the growing complexity of the VPH resources. Only a semantic approach can overcome all communication problems occur between heterogeneous resources in a huge and distribute environment.

Moreover, the provided example, even if trivial, shows how the user goal is achieved only thanks to a service composition operated by a broker, that is IRS-III.

4 LHDL DATA COLLECTION

The Living Human Digital Library will be populated during the LHDL project by an intensive data collection programme managed by the Université Libre de Bruxelles (ULB) and the Istituti Ortopedici Rizzoli (IOR).

ULB will select two cadavers from those available within their donor programme. The dissection programme commenced with the body of an adult male, who had no known abnormality or disease of the musculoskeletal system. The second specimen will be a female. The selected cadavers will be processed as soon as received from the donor programme, and the standard protocol for embalming a whole cadaver is completed.

The data collected from these two full bodies will be complemented by those collected from a variety of bone segments that will also be included in this study. All bones will have a complete clinical record for the donor and, will as broadly as possible, cover the entire age range, both genders, and both healthy and pathological bones; they will, however, be free of major anatomical defects, especially signs of fracture or damage in the region under investigation. The isolated bone specimens will be obtained through various international donor programmes, such as that operated by the International Institute for Advancement of Medicine[17].

At the Movement Analysis Laboratory of IOR, many subjects will be collected and analysed according to an original protocol for whole-body movement data. Both normal (i.e. unaffected by any relevant pathology and included to build control sub-populations), and pathological (i.e. suffering from forms of disease affecting the locomotor system or treated by a relevant surgery) will be analysed. The general control population will cover the largest possible range of age, body weight and height, for sub-populations of subjects to be arranged eventually according to the specific final interest of the analyses. The pathological population will cover total hip, knee and ankle replacements, cerebral palsy, and possibly other typical orthopaedic patients. It is planned to analyse these both immediately prior to their operation and close to typical follow-ups, i.e. 6, 12, 18, 24 months post-operatively. For these patients, a criterion for inclusion will also be the ability to walk without crutches or similar form of walking support, as these limits considerably the visibility of the external markers. The following data will be collected:

- **Whole Body level** (in-vitro)
  - Whole body CT scan
  - Whole body MRI
- **Whole Body level** (in-vivo)
  - Whole body movement data
  - Ground Reaction forces
  - Muscles activity
- **Organ level** (in-vitro)
  - Joint high-resolution CT scan
  - Joint passive kinematics
  - Quantitative anatomy
  - Whole-bone stiffness
  - Strain distribution in whole bones
  - Whole-bone strength
- **Tissue level** (in-vitro)
  - Muscle sarcomere length
  - microCT scans of cancellous bone
  - Mechanical properties of cortical bone
  - Mechanical properties of cancellous bone
- **Sub-tissue level** (in-vitro)
  - Ash density of bone
  - Non-collagen protein content
  - Microhardness of bone
  - Collagen orientation
  - Chemical composition of bone

Under the heading of quantitative anatomy, we include a number of quantitative indicators obtained through dissection, including anatomical landmarks and anatomical frames, as well as the coordinates of the areas of the muscle origins and insertions, the spatial path of the muscle/tendon fibres, the wrapping points, the superficial pennation angles, and the weight and volume of all major muscles.

5 LHDL DATA PROCESSING AND MODELLING

All of these data and measurements will be processed to extract further information and to develop predictive models of the biomechanics of the musculoskeletal apparatus. At the whole body scale, we shall develop a multiscalar volume by the spatial registration of all imaging modalities and of all measurement datasets; a full 3D model of the skeleton, by segmentation of CT images (supine pose), complete with all skeletal landmarks digitised during dissection; the same model, but oriented in the standing pose using motion capture data from volunteers; the line of action of principal muscle bundles; complete kinematics models of the major joints. From these data, it will be possible to develop inverse dynamics musculoskeletal models that take certain motion data as input and then predict muscle forces – for recent review of the methods see[18].

At the organ level, we shall focus mostly on bones. For each major bone, we shall define the joint reference systems, and the organ reference system, so as to connect the findings at the whole body scale to those at the organ level. We shall then produce finite element meshes of many major bones and use the Bonemat software to map the non-homogeneous tissue properties on to them[19]. These models will be used to predict the stresses and strains induced at the continuum level, given the joint and muscle forces acting on the bone. Last, but not least, we shall reference, at the organ level, the position of the tissue biopsies that will be used to determine all tissue-level properties.

These bone biopsies will also modelled by segmenting their microCT datasets and generating models able to predict the stresses and strains at the tissue level. Besides the "voxel meshes" method[20], we shall also explore alternative methods to obtain accurate tissue-level predictions.

These tissue-level results will be placed in correlation with the data collected at the cell scale, in order to explore the mechano-biology implications at that scale. One possibility is to develop bone remodelling algorithms that operate at the cell scale, and to propagate the predicted changes in the bone properties back to the organ level, so as to see how a particular example of bone remodelling would change certain functional attributes, such as the risk of fracture under sudden load.

6 INTERACTIVE VISUALISATION OF MULTISCALE DATA

One particular problem that is currently being investigated is how to provide an interactive visualisation environment of multiple datasets, as users of Data Manager will typically expect, when these datasets are defined at drastically different temporal or dimensional scales. For this purpose, we have generated two benchmark dataset collections.

The first of these provides a testing condition for the dimensional scale differences. The test database contains the whole bone CT dataset of a human femur, the microCT dataset of a biopsy of both cancellous and cortical bone taken from a femur, the nanoCT dataset of single trabeculae of cancellous bone, and a confocal microscopy (Figure 6) reconstruction of a fragment of cortical bone. Overall, these datasets span from metres to nanometres, providing a challenging condition for any visualisation environment. Of course, it would be trivial to visualise each dataset independently, but what we want is to retain the spatial information that links one dataset to the other, with all the difficulties that this implies. A preliminary prototype will explore an approach based on one-way exploration and upscale markers. Essentially, when we visualise the whole bone dataset, we also see some iconic markers indicating the point in space where biopsies have been taken; if I click on one of the markers, the system zooms in and visualise the datasets at that lower dimensional scale.

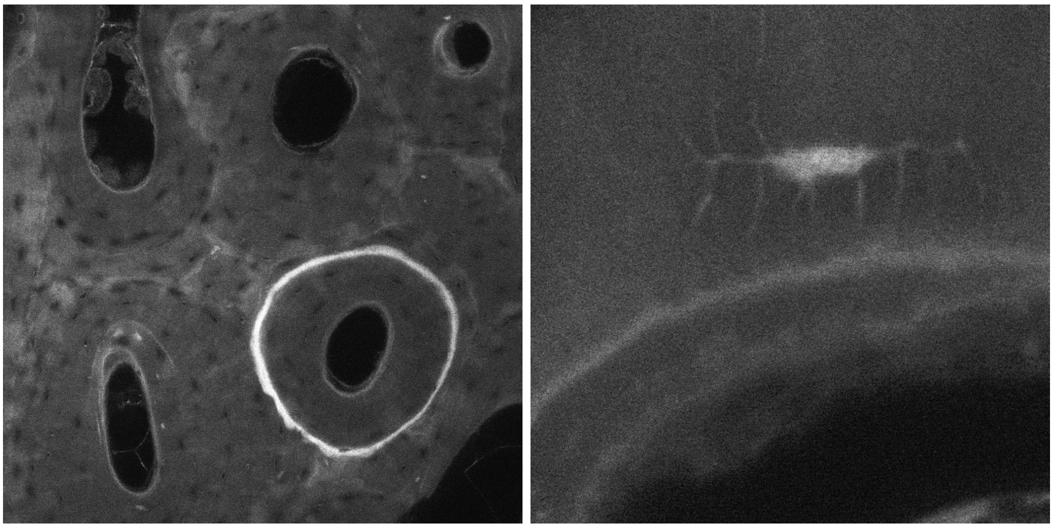

Figure 6. Confocal microscopy images of human cortical bone. Left: 20X image, where multiple osteons are visible. Right: 300X images, where a single osteocytarian lacunae and the relative canaliculi are visible

For the temporal multiscale, we have developed a benchmark dataset using a recording of an EMG array[21], provided by the courtesy of Prof. Merletti, of the Politecnico di Torino. This dataset records the propagation of the excitation in one superficial muscle of the back during a torsion movement. This signal is inherently multiscale.

At the highest scale, we observe the total excitation of the muscle during the loading-unloading exercise, which lasts around 40 seconds. If we zoom in, we can see how the signal builds up over a scale of 2-3 seconds. If we zoom down to tenths of seconds, we can see how the excitation signal is actually made of “trains” of impulses. Lastly, if we zoom down to the hundredths of seconds, we can observe how the excitation wave travels along the muscle with a given velocity (Figure 7).

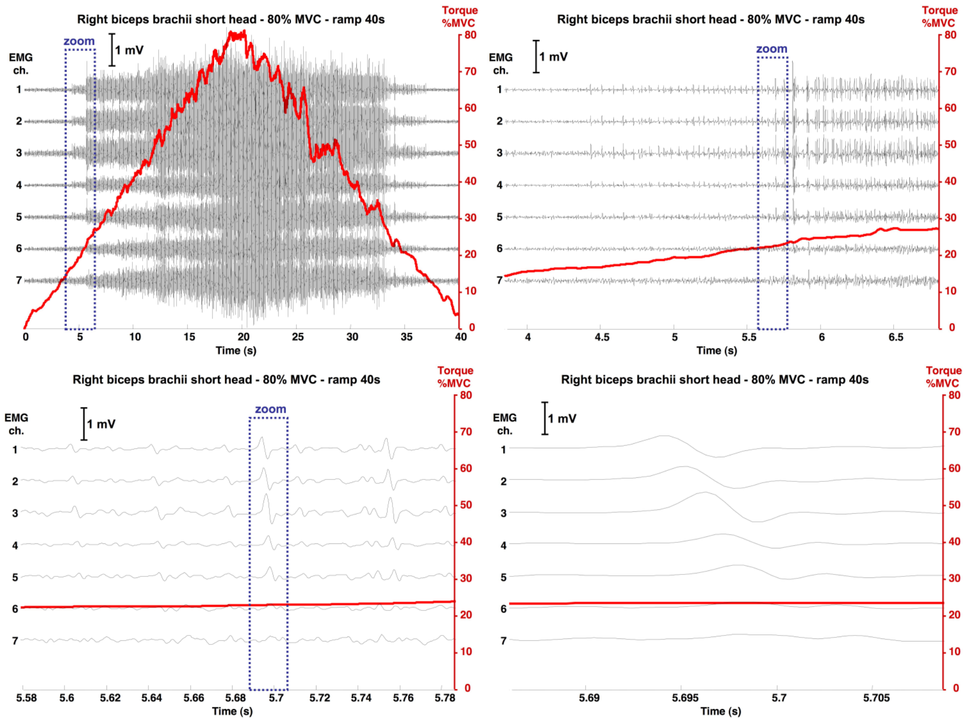

Figure 7. Multiple zooms on an EMG array recording of the right biceps brachii during an isometric force ramp. The four zooms outline the multiscale nature of the signal, which shows relevant features at various scales from seconds to milliseconds, as force changes scale from many N to mN. (Illustrations courtesy of Prof. Roberto Merletti, Lab. for Engineering of the Neuromuscular System, Department of Electronics, Politecnico di Torino, Italy)

In this case, we are working on a prototype interface that visualises relevant features at a certain time scale. So, when played at real time, the signal would progressively change colours of a muscle model to visualise the force build-up. If we slow down, we shall see how the signal is actually made of multiple bursts, and if we further slow down, we should see how the excitation wave propagates across the muscle length.

7 CONCLUSIONS

This chapter provides a preliminary description of a system currently under development. While the general vision of the system is already clear, some aspects of the implementation remain to be defined, and some of the research challenges that are emerging will probably not find a complete solution within the time span of the LHDL project. Nevertheless, we felt it important to report the work even at such an early stage, because we consider that the LHDL could become one of the most important resources for those involved with the biomechanics of the musculoskeletal apparatus. In this sense, we hope that the interested reader will contact the authors, in order to suggest complementary research projects that can put, to the best and earliest use, the infrastructure and the data collected during LHDL.

8 ACKNOWLEDGEMENTS

LHDL is a collaborative research project partly supported by the European Commission, DGINFSO H.1: ICT for Health Unit, as part of the Six Framework Program for research and technological development (FP6).

9 REFERENCES

[1] S. Judex, M. Monaghan, C. Rubin, and A. Dhundale, "Altered Mechanical Demand of the Skeleton Broadly Changes Transcriptional Activity," *J Biomech*, **39**, S22, (2006).

[2] J. B. Bassingthwaighte, "Design and strategy for the Cardionome Project," *Adv Exp Med Biol*, **430**, 325-39, (1997).

[3] Various Authors, "Seeding the EuroPhysiome - A Roadmap to the Virtual Physiological Human," STEP Coordination Action # 027642 (2006).

[4] http://www.biomedtown.org/.

[5] http://www.biomedtown.org/biomed_town/lhpsquare/lhpwarehouse/.

[6] M. Viceconti, C. Zannoni, D. Testi, M. Petrone, S. Perticoni, P. Quadrani, F. Taddei, S. Imboden, and G. Clapworthy, "The multimod application framework: A rapid application development tool for computer aided medicine," *Comput Methods Programs Biomed*, **85**, 138-151, (2007).

[7] S. Van Sint Jan, "Introducing Anatomical and Physiological Accuracy in Computerized Anthropometry for Increasing the Clinical Usefulness of Modeling Systems," *Critical Reviews of Physical Medicine and Rehabilitation*, **17**, 247-249, (2005).

[8] A. Leardini, L. Astolfi, S. Fantozzi, M. Viceconti, M. G. Benedetti, and F. Catani, "Advanced multimodal visualisation of clinical gait and fluoroscopy analyses in the assessment of total knee replacement," *Comput Methods Programs Biomed*, **79**, 227-40, (2005).

[9] L. Cabral, J. Domingue, S. Galizia, A. Gugliotta, B. Norton, V. Tanasescu, and C. Pedrinaci, "IRS-III: A Broker for Semantic Web Services based Applications," presented at 5th International Semantic Web Conference (ISWC2006), Athens, USA, (2006).

[10] http://en.wikipedia.org/wiki/Grid_computing.

[11] http://en.wikipedia.org/wiki/Service-oriented_architecture.

[12] http://www.w3.org/2002/ws/.

[13] T. Berners-Lee, J. Hendler, and O. Lassilla, "The Semantic Web," *Scientific American*, **284**, 34-43, (2001).

[14] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," *Knowledge Acquisition*, **5**, 30-35, (1993).

[15] http://www.wsmo.org/TR/d2/v1.3/.

[16] E. Motta, *Reusable Components for Knowledge Models: Principles and Case Studies in Parametric Design Problem Solving*. Amsterdam: IOS Press, (1999).

[17] http://www.iiam.org.

[18] A. Erdemir, S. McLean, W. Herzog, and A. J. van den Bogert, "Model-based estimation of muscle forces exerted during movements," *Clin Biomech*, **22**, 131-154, (2007).

[19] F. Taddei, E. Schileo, B. Helgason, L. Cristofolini, and M. Viceconti, "The

material mapping strategy influences the accuracy of CT-based finite element models of bones: An evaluation against experimental measurements," *Med Eng Phys*, pp. Epub ahead of print, PMID: 17169598, (2006).

[20] M. Viceconti, L. Bellingeri, L. Cristofolini, and A. Toni, "A comparative study on different methods of automatic mesh generation of human femurs," *Med Eng Phys*, **20**, 1-10, (1998).

[21] D. Farina, E. Fortunato, and R. Merletti, "Noninvasive estimation of motor unit conduction velocity distribution using linear electrode arrays," *IEEE Trans Biomed Eng*, **47**, 380-8, (2000).