OntoVIP: An ontology for the annotation of object models used for medical image simulation.

Bernard Gibaud, Germain Forestier, Hugues Benoit-Cattin, Frédéric Cervenansky, Patrick Clarysse, Denis Friboulet, Alban Gaignard, Patrick Hugonnard, Carole Lartizien, Hervé Liebgott, et al.

OntoVIP: an ontology for the annotation of object models used for medical image simulation

Bernard Gibaud[1], Germain Forestier[2], Hugues Benoit-Cattin[3], Frédéric Cervenansky[3], Patrick Clarysse[3], Denis Friboulet[3], Alban Gaignard[4], Patrick Hugonnard[5], Carole Lartizien[3], Hervé Liebgott[3], Johan Montagnat[4], Joachim Tabary[5], Tristan Glatard[3]

[1]LTSI - Laboratoire Traitement du Signal et de l'Image

INSERM U1099 - Université de Rennes 1

Faculté de médecine, 2 av. Pr Léon Bernard, 35043 Rennes Cedex, France

[2] MIPS - Modélisation, Intelligence, Processus et Systèmes - MIPS EA2332 - Université de Haute-Alsace,

12, rue des frères Lumière, 68093 Mulhouse, France

[3] CREATIS – Centre de Recherche et d'Applications en Traitement de l'Image et du Signal

CNRS UMR 5220 - Inserm U1044 - INSA-Lyon - Univ. Lyon 1

Bâtiment Blaise Pascal, 7 av. Jean Capelle, 69621 Villeurbanne Cedex, France

[4] I3S - Laboratoire d'Informatique, Signaux et Systèmes de Sophia Antipolis

CNRS UMR 7271/ Université Nice Sophia Antipolis

2000, route des Lucioles - Les Algorithmes - bât. Algorithm B, 06903 Sophia Antipolis Cedex, France

[5] CEA-LETI-MINATEC, Recherche technologique

17, rue des Martyrs, 38054 Grenoble Cedex 09, France

Corresponding author: Bernard Gibaud

Email: bernard.gibaud@univ-rennes1.fr

Phone : +33 2 23 23 45 90

Abstract **This paper describes the creation of a comprehensive conceptualization of object models used in medical image simulation, suitable for major imaging modalities and simulators. The goal is to create an application ontology that can be used to annotate the models in a repository integrated in the Virtual Imaging Platform (VIP), to facilitate their sharing and reuse. Annotations make the anatomical, physiological and pathophysiological content of the object models explicit. In such an interdisciplinary context we chose to rely on a common integration framework provided by a foundational ontology, that facilitates the consistent integration of the various modules extracted from several existing ontologies, i.e. FMA, PATO, MPATH, RadLex and ChEBI. Emphasis is put on methodology for achieving this extraction and integration. The most salient aspects of the ontology are presented, especially the organization in model layers, as well as its use to browse and query the model repository.**

Keywords : Simulation; ontologies; semantic web; ontology web language; medical imaging

1. Introduction

Medical imaging has become a very rich source of information which plays a major role in diagnosis, therapy and patient follow-up. Several imaging modalities, e.g. Computed Tomography (CT), Magnetic Resonance Imaging (MRI), Ultrasound (US) and Positron Emission Tomography (PET), allow exploring and imaging various facets of the morphology and physiology of a living body, at various spatial and temporal resolutions. The progress of medical imaging will certainly continue and one can foresee that all this imaging data will be used in the future to build some sort of digital patient avatars (i.e. virtual representation) composed of a set of personalized and integrated models representing anatomical, physiological and pathophysiological aspects of the organism. Such avatars could be used to test and compare various therapeutic approaches, to predict their outcome, and thus contribute to decision making. However, prerequisites for this to materialize are: (1) that such models are developed, something that initiatives like the Virtual Physiological Human (VPH) strongly support [1,2], and (2) that appropriate model identification methods are developed, whose function is to estimate the various model parameters from the specific patient multimodal image data.

In this context, an important issue remains: how to validate such models and associated identification methods? Medical image simulation appears as an interesting approach to this problem. It has undergone significant progress in the last ten years, with simulators being developed for many imaging modalities, e.g. SINDBAD [3] in CT, SIMRI [4], BrainWeb [5] in MR, SORTEO [6], GATE [7] in PET and FIELD-II [8] in US. This allows addressing a number of needs related both to the design and optimization of imaging equipment and the validation of image processing software. Indeed, image simulation allows generating realistic images of a virtual object, of which characteristics can be defined arbitrarily (e.g. presence of pathology, arbitrary choice of its size and location), and from which one can derive any kind of simulated image (i.e. by tuning spatial and temporal resolution, nature and level of noise in images, etc.). The key thing with this approach is that it provides a ground truth regarding image content, which enables quantitative assessment of image processing software. For instance, one can actually compare the result of a segmentation algorithm with the actual definition of the imaged object.

The Virtual Imaging Platform (VIP) (http://vip.creatis.insa-lyon.fr) provides researchers with a platform gathering several image simulators of various modalities [9]. Its major goals are: (1) to offer an easier access to simulators whose installation and use is usually perceived as very complex to potential users, (2) to gather simulators of various modalities, and (3) to deploy them on grid computing resources so that reasonable execution times can be obtained while image simulation usually requires huge computation.

This article focuses on the sharing and reuse of the models used for medical image simulation. A basic assumption of this work is that a major barrier to the wide-scale use of these techniques is the difficulty of creating realistic models that are suited to the researchers' specific needs. VIP aims at setting up a model repository to facilitate their sharing and reuse, based on a comprehensive conceptualization of those models, suitable for all imaging modalities and simulators considered in the VIP project and built according to an ontological approach. The first motivation for a such choice is the need to rely on of a semantically-rich vocabulary to annotate the models; such annotations will enable the VIP platform's users to assess whether an existing model can actually meet their specific needs, or be used as a starting point to derive from it an appropriate model. A second motivation concerns the representation of knowledge about the objects represented in a model involved in an image simulation, such as relating objects and materials to their physical properties; a particular aspect is the ability to use the same model in simulations with simulators of different modalities, which requires that the physical properties of materials and tissues be represented in consistent ways for all modalities. The third concerns the interoperability between biological

modeling software and medical image simulation software, which requires that common semantics are given to shared information (regarding anatomical structures, presence of pathology, quantities represented in the models or characterizing the tissues properties).

This paper describes the design methodology and the implementation of an ontology for medical image simulation models, tailored to the needs of integrating the SINDBAD, SIMRI, SORTEO and FIELD-II simulators in the VIP platform, but easily extensible to address the needs of other simulators in the future. This ontology, called OntoVIP is used to semantically annotate the models' files (images, meshes, etc.). It is freely available for consultation, download and reuse, both from the VIP web site[1], and from the National Center for Biomedical Ontology (NCBO) BioPortal[2]. This paper also demonstrates the added value of this ontology for visualizing the models content and querying the models repository.

This article is organized as follows. Section 2 describes the methodology used to design the ontology, with special attention to the reuse and consistent integration of relevant ontologies. Section 3 (Results) presents the ontology itself and its use in the VIP platform to browse and query the models repository. Section 4 (Discussion) positions our achievements with respect to our initial goals and motivations and provides further details on specific problems we have met and on the solutions proposed to overcome them. The paper concludes in section 5 with some perspectives opened by this work.

2. Material and methods

2.1. Modularity and integration framework

Modularity

Any ontological modeling has two complementary facets. The first relates to terminology and consists of agreeing on a vocabulary to name the entities of a particular domain. The second consists of associating formal semantics to the elements of this vocabulary, in order to reason about the corresponding classes of objects and their instances. Both facets are important with respect to the goal of capturing, sharing and processing data and knowledge in a particular domain.

The design of an application ontology for medical image simulation is a complex undertaking because of the interdisciplinary nature of this field, which lies at the crossroads of several domains: imaging, physics, biology and medicine. As stressed before, our goal was to define a vocabulary that is likely to gather consensus in this community. Therefore, it is important to rely on existing ontologies (if available and of sufficient quality) rather than developing new ones. Currently, the main challenges in ontology engineering are the reusability, scalability and maintenance of the modeled knowledge. In this respect, modularity is essential. Modularization consists is structuring an application ontology as the combination of independent ontology modules [10]. This approach emerged from the software engineering domain where several techniques exist to design software in such a way it could easily be modified and maintained. However, the field of ontology engineering has not reached this maturity yet, and delineating relatively independent ontological modules remains challenging in practice.

Use of a foundational ontology

In this interdisciplinary context it is also important to ensure that the resulting application ontology will be consistent. Therefore, we chose to rely on a common integration framework provided by a foundational ontology called DOLCE (Descriptive Ontology for Language and Cognitive Engineering) [11]. This choice is based on our important experience of this framework gained in the context of the NeuroLOG project [12,13]. DOLCE was produced during the WonderWeb project (2001-2003). It is an ontology of particulars and defines some 40 basic concepts such as for example `endurant`, `perdurant`, `quality`, `abstract`, and about 50 relations[3]. DOLCE is foundational in the sense that it provides entities and relations that are relevant for many kinds of application domains. It is also an axiomatic theory, and therefore constitutes "formal guidelines for domain modeling" as well as "a tool for making heterogeneous ontologies interoperate or merge" [11]. Of course, other foundational ontologies exist, such as Cyc, or the Basic Formal Ontology (BFO).

Using a foundational ontology provides a basic philosophical foundation but it is insufficient for designing an application ontology dedicated to a specialized domain such as medical image simulation. In particular, one needs to model various actions contributing to the generation of simulated images as well as the roles played by specific software components and by specific data sets in these processes. To address such needs we relied on the experience gained in the NeuroLOG project and especially on our expertise of a number of core ontologies extending the DOLCE basic classes, developed and maintained by the "Modélisation, Insformation & Systèmes" (MIS) Laboratory in Amiens, and especially an ontology of the domain of information and information bearing entities (called IEC for 'Inscription, Expression and Conceptualization') [14], an ontology of 'Artifacts' [15], and an ontology of 'Capacities and Functions'. We also reused several domain ontologies developed along the NeuroLOG project [12], such as the 'Dataset ontology' (used to model medical images of various modalities) and the 'Dataset processing ontology' (focusing on image processing).

2.2. Reuse of external ontologies

Ontology module extraction approach

An ontology module can be an entire ontology or a subset of an ontology containing the knowledge needed for a certain application. There are many advantages in extracting modules as a subset of an existing ontology rather than importing them entirely. Particularly, it helps to clearly define the scope of the application and avoids overwhelming the application ontology with numerous entities not directly linked to the application. Indeed, by importing several existing ontologies, the size of the ontology importing the modules could increase dramatically, which could be a problem for reasoning since the current reasoners are particularly sensitive to the size of the ontology. All extractions were made using the vSPARQL [16] and Gleen software [17], both developed by University of Washington in Seattle (more details are provided in Appendix A).

Ontologies modularization in VIP

In the VIP project, we selected several existing ontologies, namely the Foundational Model of Anatomy (FMA), the Mouse Pathology ontology (MPATH), the Phenotypic Attribute and Trait Ontology (PATO), the Radiology Lexicon (RadLex) and the Chemical Entities of Biological Interest ontology (ChEBI), and extracted modules containing the knowledge needed for our application (Table 1). They all were widely recognized ontologies, built according to the Open Biomedical Ontologies (OBO)

---

[3] Entities of the OntoVIP ontologies are in `Courier New` font; italics are used to denote entities that were imported from some external ontology

Foundry principles [18] (especially they are orthogonal, i.e. do not overlap), and available from the NCBO BioPortal. By reusing existing ontologies, we increase the compatibility with existing software and platforms. Furthermore, we ease the construction of our ontology as we can focus on modeling the concepts specific to our application.

Module Name	Origin	Nb of classes	Import/adaptation
vip-model	New module	185	Definition according to the OntoSpec methodology
vip-biological-objects	FMA v3.1 (ref [19])	865	vSPARQL extraction (Cotopies + transitive closure of relations)
vip-pathological-anatomical-object-quality	PATO V1.2 (ref [26])	84	vSPARQL extraction (Recursive children of `physical object quality` + manual pruning)
vip-pathological-object	MPATH V1.2 (ref [25])	494	vSPARQL extraction (Recursive children of `pathological anatomical entity`)
vip-contrast-agent	RadLex V3.2 (ref [27])	81	vSPARQL extraction (Recursive children of `contrast agent` + manual addition)
vip-radiopharmaceutical	RadLex V3.2 (ref [27])	49	vSPARQL extraction (Recursive children of `radiopharmaceutical` + manual addition)
vip-foreign-body-object	RadLex V3.2 (ref [27])	189	vSPARQL extraction (Recursive children of `foreign body` + manual addition)
vip-atom	ChEBI V93 (ref [28])	254	vSPARQL extraction (Recursive children of `chemical entity`)
OntoNeuroLOG-dataset-processing	OntoNeuroLOG (ref [12])	124	Existing module of OntoNeuroLOG
Dataset	OntoNeuroLOG (ref [12])	76	Existing module of OntoNeuroLOG
Medical-dataset-expression	OntoNeuroLOG	38	Existing module of OntoNeuroLOG
Core-ontology-programs-software	OntoNeuroLOG	153	Existing module of OntoNeuroLOG
OntoNeuroLOG-extension-of-DOLCE*	OntoNeuroLOG	183	Existing module of OntoNeuroLOG
DOLCE-particular	DOLCE (ref [11])	38	Existing module of OntoNeuroLOG

* this module gathers several ontology files from OntoNeuroLOG, such as 'Inscription, Expression and Conceptualization' (IEC), 'Capacities and Functions' and 'Artifacts'

Table 1: Summary of the OntoVIP ontology modules that were developed or extracted/adapted from existing ontologies.

FMA: Anatomical objects are important components of simulation object models since a large part of object models describes the organs contained in the models. One of the keys of object models sharing is to refer to the same anatomical concepts and to control the vocabulary used to describe the anatomy. We decided for this part to rely on the FMA [19] which is an evolving computer-based knowledge source about human anatomy. It is a domain ontology, which means that its goal is to be exhaustive and not linked to any specific application. Consequently, FMA is a large ontology. It contains around 80,000 concepts and more than 2.5 million relations. This can be a problem in case automatic inferences are needed. Indeed, current reasoners are not able to handle such an important number of concepts and relations.

We decided to extract a subset of the FMA relevant for our project, i.e, extracting anatomical terms which could be used in the description of object models for simulation. We started by listing anatomical terms used in the description of several

models, namely XCAT [20], BrainWeb [21], ADAM [22] and Zubal [23]. In addition to these publications we used the model description files in their native formats. From these resources, we extracted around 200 terms.

We decided to extract the list of identified concepts along with their upper cotopy [24]. Cotopy extraction consists in keeping a concept and all its parents, recursively until the root of the ontology is reached. This choice was made to increase the number of anatomical concepts covered by our ontology and to create a multi-level representation. Indeed, we make the assumption that the user could be interested in more generic concepts than the ones described in the existing models. Especially, this may be helpful for relating anatomical structures to the physical parameters of the tissues; actually the latter will likely be defined in reference to broad classes of tissues rather than detailed anatomical structures. We avoided including more concepts (as the lower cotopy, i.e. all the children of a concept) to avoid overwhelming the user with too many precise terms. However, we also included all the concepts linked to the set of initial concepts by the relations: [`fma:regionalPart`][`fma:regionalPartOf`]. We extracted the transitive closure of these relations as well as the cotopies of the linked concepts. The extracted ontology with the identified concepts and their upper cotopies and the transitive closure of the relations contains roughly 280 concepts. We also included the FMAID, synonyms and textual description of the extracted concepts when they were present. Finally, we also kept the Freesurfer[4] ID's for further compatibility with segmentation results obtained using this very popular neuroimaging software.

MPATH: We extracted the pathological structures from the Mouse Pathology ontology [25]. The main reasons for this choice are firstly the relevance of this ontology for most animal species concerned by medical image simulation (including humans), and secondly that it is of limited complexity, compared, e.g., to SNOMED CT, which could have been an alternative. Our selection of the concepts of interest was less restrictive than the one concerning anatomical terms, as we had less information about the potential needs in pathological structures. Because the size of the MPATH ontology is reasonable, we decided to extract recursively all the children starting from the concept `pathological anatomical entity`. The extracted module contains 494 concepts.

PATO: The concepts extracted from MPATH are pathological structures (e.g. `neoplasm`). However, we also wanted to allow the possibility to represent anatomical structures affected by a quality transforming its nature to a pathological one (e.g. a `fatty heart` modeled as `heart` and `hasForQuality` some `fatty`). We started by extracting all the children of the concept `physical object quality` from the Phenotypic Attribute and Trait Ontology (PATO) ontology [26]. We pruned the concepts manually to remove the concepts which were not interesting for our application. We kept 84 concepts in this module.

RadLex: RadLex is a controlled terminology source of radiology terms for radiology practice, education, and research [27]. It is an ongoing project of the Radiological Society of North America (RSNA). We extracted three different modules. The first one contains the contrast agents, we extracted all the children of the concept `contrast agent`, the second contains the radiopharmaceuticals, we extracted all the children of the concept `radiopharmaceutical`, and the last one contains the

---

[4] freesurfer: http://ftp.nmr.mgh.harvard.edu/

foreign objects (e.g. needle), we extracted all the children of the concept `foreign object`. The modules contain respectively 81, 49, and 189 concepts. Some concepts were added manually afterwards to address additional user needs of concepts that were not present in RadLex.

ChEBI: We extracted the list of atoms from the Chemical Entities of Biological Interest (ChEBI) [28] ontology downloaded from the project website: http://www.ebi.ac.uk/chebi. These concepts were needed to model the chemical composition of the objects. We extracted all the children of the concept `chemical entity`, corresponding to 254 concepts.

II.3. Creation of new ontology modules

Several aspects of the domain of discourse of medical image simulation were not covered by existing ontologies. So we had to develop the corresponding modules, following the OntoSpec methodology. OntoSpec assists the ontological knowledge modeling step, "upstream of the formal representation and knowledge implementation steps" [29]. According to this methodology each new ontology component is first defined as a textual document of semi-formal nature, capturing rich semantics. In particular, each new conceptual entity (concept or relation) is characterized by its essential properties, such as its subsumption link with parent concepts or relations, or "existential restrictions" involving some relations and classes of the ontology. This semi-formal part is usually completed by an informal definition and together with relevant examples, citations or references to external documentation. The semi-formal part of OntoSpec documents was then translated (in whole or in part) into OWL-DL, a specific dialect of the Web language for ontologies (OWL) based on Description Logics (DL), and used in the implementation of the VIP platform.

3. Results

3.1. Ontology of Medical image simulation object models

The ontology of medical image simulation object models[5] allows precise semantics to be associated to the data files that compose a model. Depending on the different simulators, the models have a very diverse structure; the models' geometry may be represented as images (i.e. voxel maps), or meshes (i.e. object surfaces) depicting the spatial extent of the objects composing the scene, and representing, for example: organs, organ parts, lesions, contrast agents, tracers, or foreign bodies. Besides, all image simulation object models involve, directly or indirectly, some representation of the physical parameters of the objects or tissues represented in the scene. However, in many cases the nature of the tissues and the nature of their physical parameters are not explicit, or represented in ways that are not consistent across the various simulators.

The proposed ontology aims at covering and modeling consistently all kinds of medical image simulation object models, especially with regards to the different imaging modalities (CT, MR, PET, US). This results from the effort of abstraction made in analyzing the specific needs of the various simulators to be integrated in the VIP platform.

Organization in layers

---

[5] `model` is used for short, but the complete name of this entity is `medical image simulation object model`

Each `model` is composed of one or more `model layers` that provide information about the different kinds of object of a model (Fig. 1). Both entities are propositional contents and therefore modeled as *propositions* (from the IEC ontology). Each `model layer` belongs to one `model` only (relation *isAProperPartOfAt*). Basically two kinds of `model layers` can be distinguished: (1) `object layers`, which depict an object map (via label values), and (2) `values layers` which depict the map of a given physical quantity, for example the spatial distribution of T1 relaxation times throughout the object. Object layers are categorized into 5 classes which depend on the common class of objects represented in this layer, namely: (1) `anatomical object layer` for anatomical objects, (2) `external agent object layer` for contrast media or radioactive tracers, (3) `foreign body object layer` for external objects such as implants, tubes, or electrodes, (4) `pathological object layer` for lesions or pathological tissues and (5) `geometrical phantom object layer` for phantoms, such as test objects used to measure spatial resolution of imaging systems (Fig. 1).
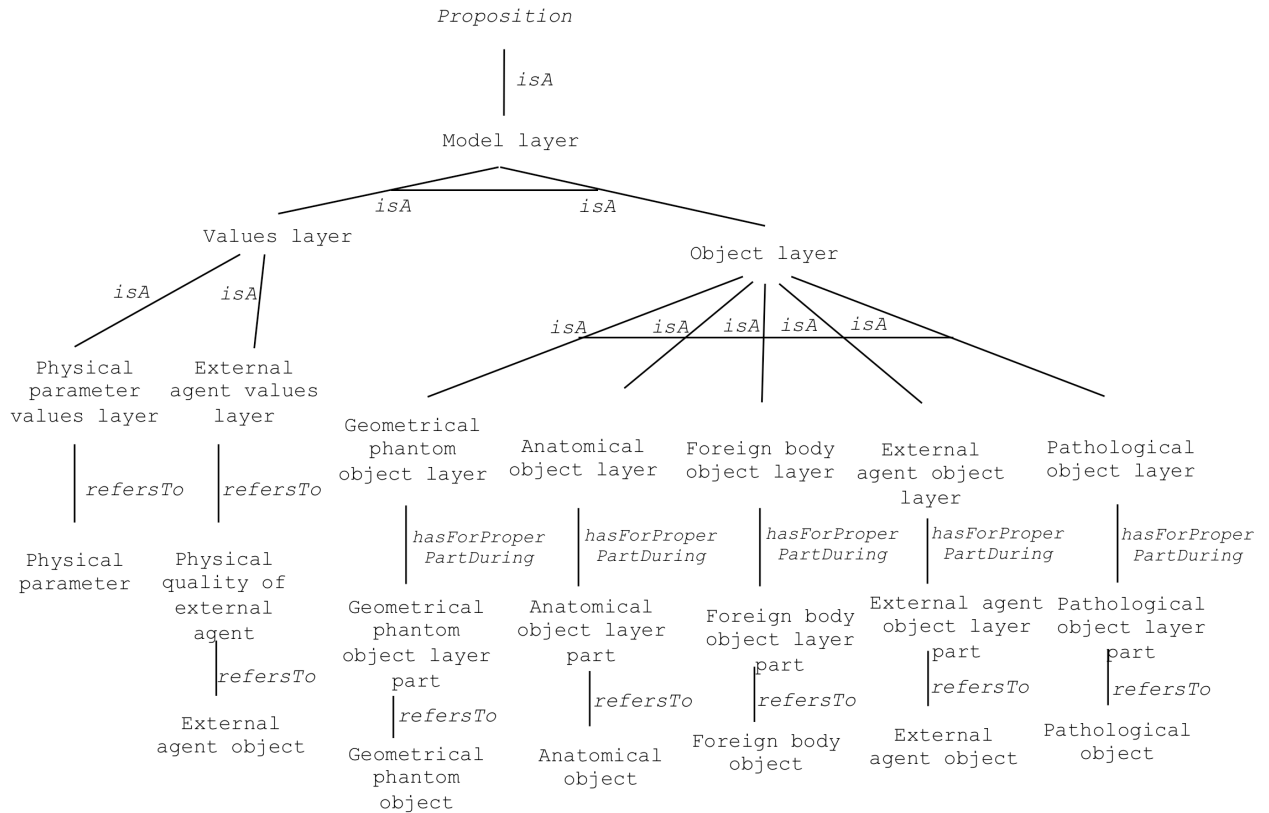

Figure 1. Taxonomy of model layers: relations are represented using lines; horizontal lines crossing subsumption links denote disjoint child classes

Each `object layer` is composed of one or more `object layer parts`, which can be either an `object layer part voxel` or `object layer part mesh`. The former is denoted by a common label value in a pixel or voxel map (data property `hasForLabelInModel`) whereas the latter is characterized by a priority level, allowing the order of inclusion of the objects surfaces to be managed (data property `hasForPriority`). `Object layer parts` refer to real objects (using the

refersTo object property that associates any kind of object to a `representational object`). `Object layer parts` are modeled as `propositions` and also further categorized into, e.g., `anatomical object layer part`, or `external agent object layer part`, which refer to (using `refersTo`) `anatomical objects`, `external agent objects`, respectively. Fig. 2 provides an illustrative example of a model combining `values layers` (more precisely `physical parameter values layers`) and `object layers` (an *anatomical object layer* and a *pathological object layer*).

Geometry files

Semantic annotations
(organisation in object layers and values layers)

`physical parameter values layer` instance,
referring to a `T1` instance (magnetic property)

`physical parameter values layer` instance,
referring to a `T2` instance (magnetic property)

`physical parameter values layer` instance,
referring to a `proton density` instance (magnetic property)

`anatomical object layer` instance,
with 8 `anatomical object layer part` instances, referring to:
- a `Cerebrospinal fluid` instance
- a `Gray matter of neuraxis` instance
- a `White matter of neuraxis` instance
- a `Fat body` instance
- a `Portion of muscle tissue` instance
- a `Skin of head` instance
- a `Skull` instance
- a `Portion of connective tissue` instance

`pathological object layer` instance,
with 1 `pathological object layer part`, referring to:
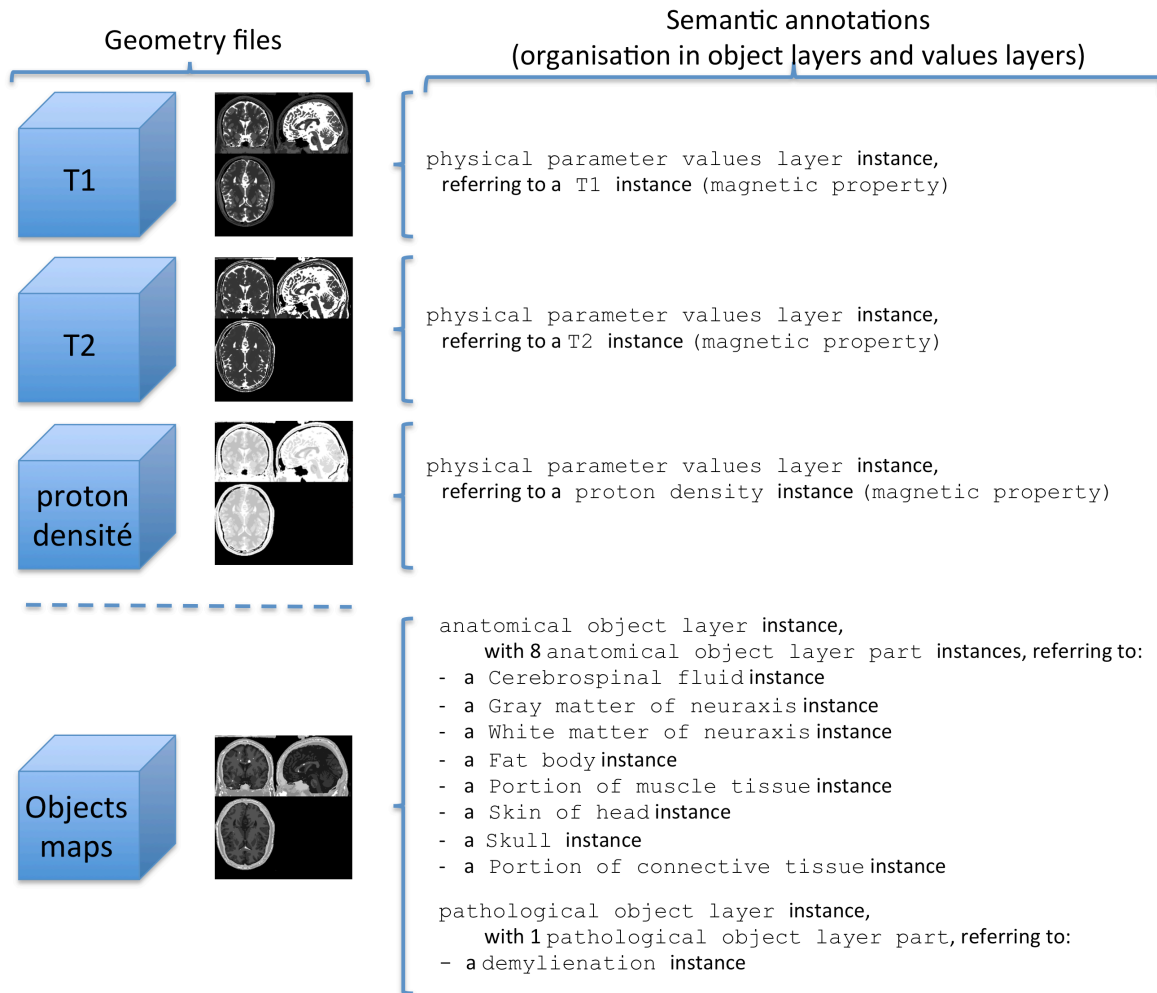- a `demylienation` instance

Figure 2. Illustrative example based on the "BrainWeb-Severe-Multiple-Sclerosis-with-Maps" model. The left part of the figure shows the geometry files and the right part the semantic annotations that make the content of these geometry files explicit. This model is composed of three `physical parameter values layer` instances depicting the T1, T2 and proton density values throughout the scene, and of two `object layer` instances, an `anatomical object layer` instance (composed of eight `anatomical object layer part` instances) and a `pathological object layer` instance (composed of a single `pathological object layer part` instance). The `anatomical object layer` and `pathological object layer` instances refer to (`isStoredinFile` data property) the same physical files (i.e. they share a common objects map), whereas the three `physical parameter values layer` instances refer to different files.

Link with physical properties

The link with physical properties is essential since the latter are used by the simulators to simulate the physical interaction phenomena involved in image acquisition processes, processes that are specific of each imaging modality. `Physical properties` may either be associated to `object layer parts`, when `object model layers` are used, or associated to `values layers`, that by definition represent the distribution of a physical property.

In the first case a `mathematical distribution of physical quality` is associated to each `object layer part`. A `mathematical distribution of physical quality` is a *proposition* which describes a distribution of some physical parameter through its essential `mathematical distribution parameters` (object property `hasForParameter`). This allows representing in a homogeneous way various situations in which actual parameter values may be, e.g., constant throughout the object (`constant distribution`, `hasForConstantValue` exactly 1 `constant value`) or follow a Gaussian distribution (`Gaussian distribution`, `hasForMean` exactly 1 `mean`; `hasForStandardDeviation` exactly 1 `standard deviation`). Four distributions were defined to address the needs of SORTEO, SINDBAD, SIMRI and FIELD-II simulators (see Table 2).

Mathematical distribution	Mathematical distribution parameter
`constant distribution`	`constant value`
`uniform distribution`	`lower bound` `upper bound`
`gaussian distribution`	`mean` `standard deviation`
`K distribution`	`scale parameter` `shape parameter`

Table 2: Subclasses of `mathematical distribution of physical quality` and their associated `Mathematical distribution parameters`.

Other distributions may of course be added in the future if necessary. Each instance of `mathematical distribution of physical quality` *refersTo* some instance of *physical quality*.

The taxonomy of *physical qualities* is presented Fig. 3. Similarly, each instance of `values layer` *refersTo* an instance of *physical quality*.
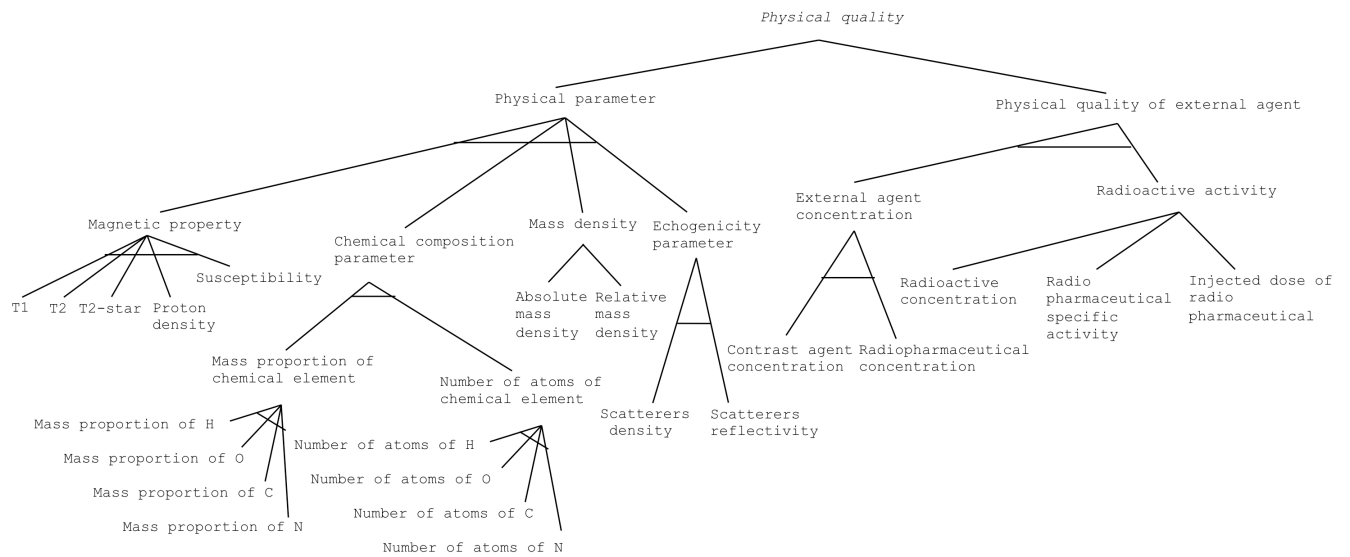
Figure 3. Taxomomy of physical qualities (that were not present in PATO); lines represent subsumption links; lines crossing subsumption links denote disjoint child classes

Temporal aspect

Models may represent time-varying phenomena. This is obviously important to cover the domain of dynamic medical imaging which concerns all imaging modalities, and is critical to image moving organs such as heart and lungs. It is also important in order to meet the requirements related to evolutive pathological processes, e.g. tumor growing or shrinking. OntoVIP addresses this need by associating to the model layers a temporal reference. Two temporal scales are considered: the notion of `instant` addresses the dynamic imaging issue and allows describing in a discrete way all the successive states of the imaged object (`model layer` *refersTo* `instant`). The notion of `time point` concerns the modeling of dynamic processes which exploration would rely - in real life - on distinct examinations several days or several months apart. Of course, both aspects can be combined to allow analyzing dynamic phenomena that are themselves of dynamic nature. This is modeled using a part-whole relationship between `instant` and `time point` (an `instant` *isAProperPartOf* exactly 1 `time point`).

Categorization of models

Models can be categorized along four main semantic axes which aim at facilitating the query of models from our repository. The first axis introduces a distinction between static and dynamic models (`static object model` versus `dynamic object model`). The second axis highlights the nature of the represented object, for example `biological object models` represent biological objects whereas `geometrical phantom object models` represent geometrical objects such as those used to measure the spatial resolution. Both classes are disjoint. The third semantic axis conveys information about the objects that are present in the model: for instance, an `anatomical object model` includes at least one `anatomical object layer` (object property *hasForProperPartDuring*). Similarly, we introduced classes for `pathological object models`, `non pathological object models`, `object models with external agent`,

`object models without external agent`, etc. This taxonomy of object models is depicted Fig. 4. Lastly, a fourth axis characterizes models that are compatible with some class of image simulation actions. This property is modeled as a *capacity* (which is a *state borneBy* a `model`). For example, a `model` is a `CT-simulation compatible model` if it bears some `CT simulation capacity`, defined as a *capacity to enable action*, which *enablesToFulfil* some `role of model in CT simulation`. In practice, membership to the `CT simulation compatible model` class is inferred using a dedicated rule expressing a condition about the presence of those physical parameters that are required for the corresponding class of simulators.
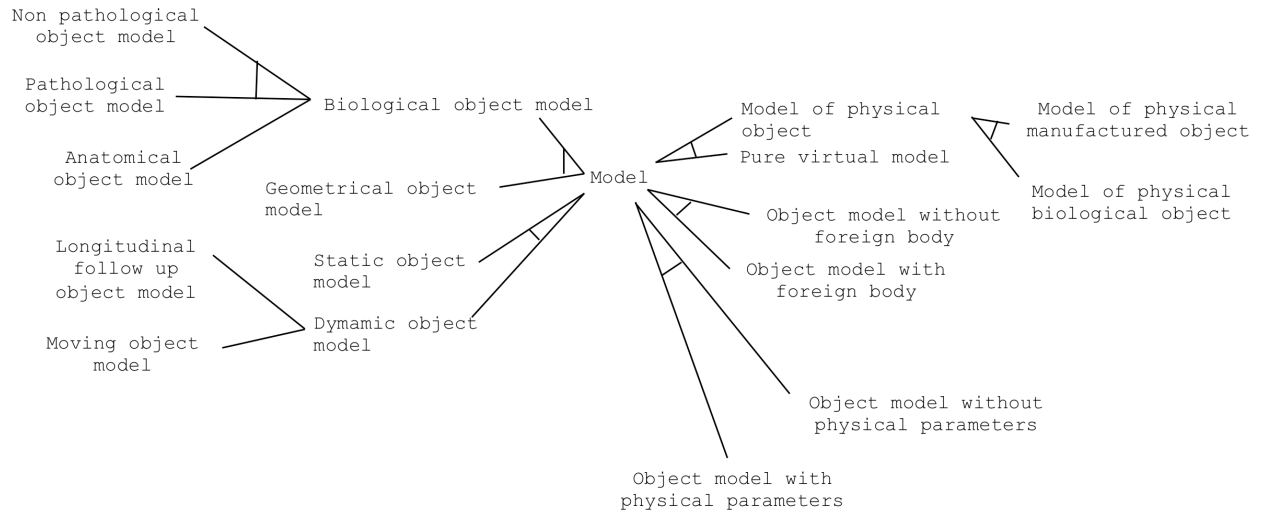
Figure 4. Taxonomy of models: lines represent subsumption links; lines crossing subsumption links denote disjoint child classes

3.2. Use of the ontology in the VIP platform

Annotation of medical object models: The VIP platform manipulates medical object models (which are the input of the simulator) by using annotations. An object model is composed of several data files representing the content of the model (geometry and physical parameters). Annotating a model consists in creating a Resource Description Framework (RDF) file containing instances of the concepts of our ontology and describing the structure and the content of the object model. Consequently, a fully annotated model is composed of a set of data files (containing geometry and physical parameters) and a RDF file describing the semantic content of these data files. When users connect to the platform, they can browse a catalog of existing models that have been previously created, and use them directly to launch simulations. Alternatively, they can also create their own models but they have to create the annotations describing their content. A dedicated graphical user interface is available in the platform to help annotating new models. A drag-and-drop system is used to upload the data files related to a model and dialog windows are used to guide the users in the annotation of the data files. For example, when a user uploads a file containing the geometry of an object, a dialog window helps him searching the different objects available in the VIP

ontology (e.g. anatomical object, pathological object, etc.). This interface indexes all the terms (i.e. labels of the concepts) available in the different modules of the ontology. It is possible to make fuzzy search among them, the result being a list of terms ordered by their degree of matching. It is also possible to specify the scope of the search in order to focus the query.

Use of reasoning: By using a semantic approach, we were able to use inference and reasoning to construct and manipulate models in the platform. The annotation process uses the subsumption information of the ontology to automatically infer the type of the object that the user selects during the annotation. Consequently, the user only states that the object referred to in a geometrical file (e.g. brain.vtp) is, e.g., a `Brain` (by querying ontology's terms), and the interface automatically infers what kind of `object layer` and `object layer part` are concerned (i.e. an `anatomical object layer` and an `anatomical object layer part`, respectively).

Reasoning is also used after the annotation in order to infer additional information on the models. This allows to automatically categorize models using semantic rules (i.e. `anatomical object model`, `pathological object model`, etc.). Table 3 lists the rules currently used in the implementation.

Inference rule*	Inferred property
Anatomical model rule	`vip-model:anatomical-object-model`
Pathological model rule	`vip-model:pathological-object-model`
Non pathological model rule	`vip-model:non-pathological-object-model`
External agent model rule	**`vip-model:object-model-with-external-agent`**
Non external agent model rule	**`vip-model:object-model-without-external-agent`**
Foreign body model rule	**`vip-model:object-model-with-foreign-body`**
Non foreign body model rule	**`vip-model:object-model-without-foreign-body`**
Follow-up model rule	`vip-model:longitudinal-follow-up-object-model`
Moving model rule	`vip-model:moving-object-model`
Static model rule	`vip-model:static-object-model`

* available at: http://vip.creatis.insa-lyon.fr/ontovip/modules/semanticAxes.rules

Table 3: List of inference rules used to categorize object models.

As an example, the following rule infers membership to the `anatomical object model` class if at least one `anatomical object layer` is present in the model:

```
[anatomical:(?m rdf:type vip-model:medical-image-simulation-object-model)
(?m particular:has-for-proper-part-during ?l)
(?l rdf:type vip-model:anatomical-object-layer)
-> (?m rdf:type vip-model:anatomical-object-model)]
```

The consistency of the annotations under construction is also checked using a reasoning engine. This feature is a strong advantage of our approach as the user can benefit from the semantic of the data to have feedback on the annotations. The compatibility of the models with imaging modalities is also inferred automatically. A model compatible with a modality (e.g. MR) is a model containing enough information to run a simulation of this modality using this model as input. For example, to

run an MR simulation, the physical parameters T1, T2, proton density must be defined for each object of each layer of the model. This information is inferred automatically from the availability of appropriate instances of mathematical distribution of physical qualities (i.e. associated to those qualities required for a given modality). All these automatic inferences are made possible thanks to our semantic modeling.

Implementation: The VIP platform was created using the Google Web Toolkit (GWT). To design the annotation software, we used Jena, which is a library developed by the Apache Foundation (http://jena.apache.org/). One of the main challenges was to manipulate the annotations on the web platform during the creation of a model. Indeed, as the application is running inside the web-browser of the client, it is not possible to include all the needed libraries (like Jena). Consequently, all the operations consisting in creating and manipulating the annotations had to be done on the server-side while being driven by user interaction on the client-side. These constraints led us to the definition of a communication protocol between the client and the server to exchange information on annotations (Fig. 5). A simple Java objects model was designed, composed of representational objects used to send the information to display to the client (Data Transfer Objects). This approach makes it possible to iteratively create the semantic annotations on the server-side.
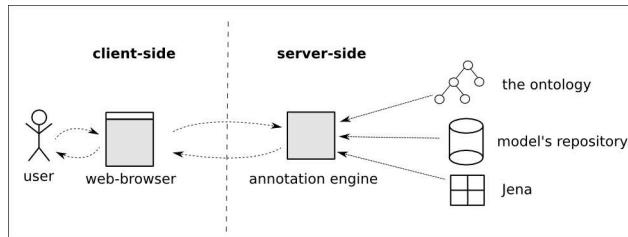

Figure 5. Implementation architecture of the annotation application

Example of model: In order to populate the catalog of models, we created several models from available online resources. We created 10 models which illustrate most of the features of our ontological representation (Table 4).

Model name	Model description	Data origin and citation
BrainWeb-Normal-with-Luts	Normal subject with - 1 anatomical object layer with 11 anatomical objects and 5 related LUTs for MR magnetic properties	BrainWeb – subject04 http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_normal_20.html (see [30])
BrainWeb-Normal-with-T1-T2-PD-maps	Normal subject with - 3 physical parameter values layers related to 3 magnetic properties (T1,T2, proton density) - 1 anatomical object layer with 11 anatomical objects	BrainWeb – subject04 http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_normal_20.html (see [30])
IrcaDB1.1-patho-with-Luts	Subject with cyst and tumors - 1 anatomical object layer with 11 anatomical objects and 5 related LUTs for MR magnetic	http://www.ircad.fr/softwares/3Dircadb/3Dircadb1/index.php

	properties - 1 pathological object layer with 2 pathological objects and 5 related LUTs for MR magnetic properties	
IrcaDB1.1-patho-with-T1-T2-PD-maps	Subject with cyst and tumors - 3 physical parameter values layers related to 3 magnetic properties (T1,T2, proton density) - 1 anatomical object layer with 11 anatomical objects - 1 pathological object layer with 2 pathological objects	http://www.ircad.fr/softwares/3Dircadb/3Dircadb1/index.php
BrainWeb-Mild-Multiple-Sclerosis-with-Luts	Subject with mild multiple sclerosis - 1 anatomical object layer with 8 anatomical objects and 5 related LUTs for MR magnetic properties - 1 pathological object layer with 1 pathological object and 5 related LUTs for MR magnetic properties	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms1.html
BrainWeb-Mild-Multiple-Sclerosis-with-Maps	Subject with mild multiple sclerosis - 3 physical parameter values layers related to 3 magnetic properties (T1,T2, proton density) - 1 anatomical object layer with 8 anatomical objects - 1 pathological object layer with 1 pathological object	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms1.html
BrainWeb-Moderate-Multiple-Sclerosis-with-Luts	Subject with moderate multiple sclerosis - 1 anatomical object layer with 8 anatomical objects and 5 related LUTs for MR magnetic properties - 1 pathological object layer with 1 pathological object and 5 related LUTs for MR magnetic properties	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms.htm
BrainWeb-Moderate-Multiple-Sclerosis-with-Maps	Subject with moderate multiple sclerosis - 3 physical parameter values layers related to 3 magnetic properties (T1,T2, proton density) - 1 anatomical object layer with 8 anatomical objects - 1 pathological object layer with 1 pathological object	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms.htm
BrainWeb-Severe-Multiple-Sclerosis-with-Luts	Subject with severe multiple sclerosis - 1 anatomical object layer with 8 anatomical objects and 5 related LUTs for MR magnetic properties	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms3.html

	- 1 pathological object layer with 1 pathological object and 5 related LUTs for MR magnetic properties	
BrainWeb-Severe-Multiple-Sclerosis-with-Maps	Subject with severe multiple sclerosis - 3 physical parameter values layers related to 3 magnetic properties (T1,T2, proton density) - 1 anatomical object layer with 8 anatomical objects - 1 pathological object layer with 1 pathological object	BrainWeb – multiple sclerosis http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_ms3.html

Table 4: Details of the origin and content of ten models from the VIP model repository.

Eight models were derived from the BrainWeb (http://brainweb.bic.mni.mcgill.ca/brainweb/) simulator's geometry files. We also created physical parameters maps allowing the models to be used as input of simulation. Two models were also created using as input 3D scanner models from a database available on the "Institut de Recherche contre les Cancers de l'Appareil Digestif" (IRCAD) web site (http://www.ircad.fr/). We present here one model (named BrainWeb-Severe-Multiple-Sclerosis-with-Maps) to illustrate the annotation file created during the annotation process. Fig. 6 presents the visualization of the annotations on the VIP platform as a tree structure representing the content of the model.
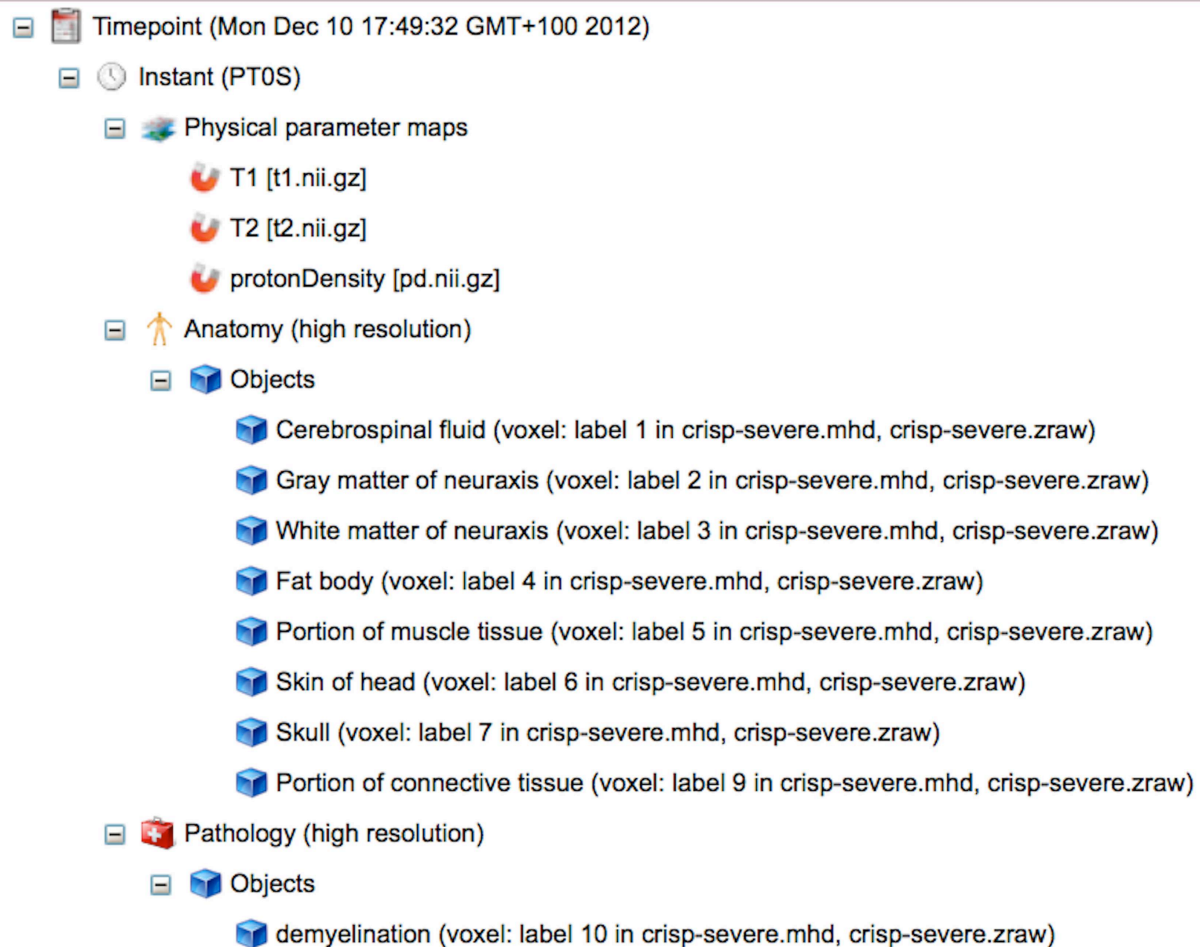
Figure 6. Representation as a tree structure of the content of the model named "BrainWeb-Severe-Multiple-Sclerosis-with-Maps".

This view is automatically built by parsing the annotation file on the server side. Fig. 7 presents an extract of the annotations file which was created using the graphical annotation tool available on the platform. The annotations were simplified for visualization purpose. These models were used during a tutorial session organized for the launching of the platform (http://vip.creatis.insa-lyon.fr/documentation/hands-on-semantics.html).

```xml
<?xml version="1.0" encoding="US-ASCII"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
```

```xml
<rdf:Description rdf:about="#medical-image-simulation-object-model-b414faa71e99442da470db6585150586">
    <rdf:type rdf:resource="#anatomical-object-model"/>
    <rdf:type rdf:resource="#pathological-object-model"/>
    <rdf:type rdf:resource="#object-model-without-external-agent"/>
    <rdf:type rdf:resource="#static-object-model"/>
    <rdf:type rdf:resource="#medical-image-simulation-object-model"/>
    <rdf:type rdf:resource="#object-model-without-foreign-body"/>
    <j.1:has-for-proper-part-during rdf:resource="#pathological-object-layer-
72ec8831474d4385a4138a906d9bba97"/>
    <j.1:has-for-proper-part-during rdf:resource="#anatomical-object-layer-
17c70b5554ba49759fc77d4d492c6f92"/>
    <j.4:has-for-model-description>Empty description</j.4:has-for-model-description>
    <j.4:has-for-model-owner>Bernard Gibaud</j.4:has-for-model-owner>
    <j.4:has-for-last-modification-date rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2012-
12-10T16:49:32.307Z</j.4:has-for-last-modification-date>
    <j.4:has-for-model-name>BrainWeb-Severe-Multiple-Sclerosis-with-Maps</j.4:has-for-model-name>
  </rdf:Description>

<rdf:Description rdf:about="#anatomical-object-layer-17c70b5554ba49759fc77d4d492c6f92">
    <rdf:type rdf:resource="#anatomical-object-layer"/>
    <j.1:has-for-proper-part-during rdf:resource="#anatomical-object-layer-part-
6d77f408730f45a0909068ab6bd5067f"/>
    <j.1:has-for-proper-part-during rdf:resource="#anatomical-object-layer-part-
fd84f83b8140474e91903017e33a3087"/>
    <j.1:has-for-proper-part-during rdf:resource="#anatomical-object-layer-part-
61181be45d544cf8bda6aa7a6c9c6afa"/>
  </rdf:Description>

<rdf:Description rdf:about="#anatomical-object-layer-part-61181be45d544cf8bda6aa7a6c9c6afa">
    <rdf:type rdf:resource="#object-layer-part-voxel"/>
    <rdf:type rdf:resource="#anatomical-object-layer-part"/>
    <j.4:is-stored-in-file rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">crisp-
severe.mhd</j.4:is-stored-in-file>
    <j.4:is-stored-in-file rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">crisp-
severe.zraw</j.4:is-stored-in-file>
    <j.2:refers-to rdf:resource="http://sig.uw.edu/fma#White_matter_of_neuraxis-
f605cf7962124731ba57fece75520165"/>
    <j.4:has-for-label-in-model rdf:datatype="http://www.w3.org/2001/XMLSchema#long">3</j.4:has-for-
label-in-model>
  </rdf:Description>

<rdf:Description rdf:about="#pathological-object-layer-72ec8831474d4385a4138a906d9bba97">
    <rdf:type rdf:resource="#pathological-object-layer"/>
    <j.1:has-for-proper-part-during rdf:resource="#pathological-object-layer-part-
3fda5cf2822e4de7888f9641a6930c8b"/>
  </rdf:Description>
```

```
<rdf:Description rdf:about="#pathological-object-layer-part-3fda5cf2822e4de7888f9641a6930c8b">
    <rdf:type rdf:resource="#object-layer-part-voxel"/>
    <rdf:type rdf:resource="#pathological-object-layer-part"/>
    <j.4:is-stored-in-file rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">crisp-
severe.mhd</j.4:is-stored-in-file>
    <j.4:is-stored-in-file rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">crisp-
severe.zraw</j.4:is-stored-in-file>
    <j.2:refers-to rdf:resource="http://vip.cosinus.anr.fr/vip-pathological-object.owl#demyelination-
872829569f8f41579d7dcdc32f33c059"/>
    <j.4:has-for-label-in-model rdf:datatype="http://www.w3.org/2001/XMLSchema#long">10</j.4:has-for-
label-in-model>
  </rdf:Description>
```

Figure 7. Simplified extract of a model annotation file

Querying the models: The models can obviously be sorted by name, author, date of modification, etc. It is also possible to use a text field to search for the presence of a specific object in the content of the models. For example, it is possible to list all the models that contain an object `Brain`.

Semantic queries are also supported to search the model in a more complex way. Indeed, the annotations corresponding to the models available in the platform are stored in a semantic repository. Thus, SPARQL queries can be run against this repository to take advantage of the semantic richness of the annotations. Currently, the SPARQL query tool is not included in graphical user interface of the platform as the users are not expected to be familiar with this language. However, we designed pre-defined queries that can be executed by a simple click. For example, the query presented in Fig. 8 displays the list of models of the repository containing anatomical objects and uses the knowledge imported from the FMA (and included in our ontology) to display the regional parts associated to the anatomical objects contained in the models. This kind of knowledge can be used for example, for query expansion. Indeed, if the user is looking for an object that cannot be found in the existing models, the system can then suggest querying the regional parts of this object, or the objects that the queried object is a part of.

```
PREFIX fma:<http://sig.uw.edu/fma#>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl:<http://www.w3.org/2002/07/owl#>
PREFIX vip-model:<http://vip.cosinus.anr.fr/vip-model.owl#>
PREFIX vipmr:<http://www.irisa.fr/visages/team/farooq/ontologies/mr-protocol-owl-lite.owl#>
PREFIX vipiec:<http://www.irisa.fr/visages/team/farooq/ontologies/iec-owl-lite.owl#>
PREFIX vipparticular:<http://www.irisa.fr/visages/team/farooq/ontologies/particular-owl-lite.owl#>


SELECT ?name ?objecttype ?regionalparts
WHERE {
  ?model rdf:type vip-model:medical-image-simulation-object-model.
  ?model vip-model:has-for-model-name ?name.
  ?model vipparticular:has-for-proper-part-during ?layer.
  ?layer rdf:type vip-model:anatomical-object-layer.
  ?layer vipparticular:has-for-proper-part-during ?layerpart.
  ?layerpart vipiec:refers-to ?object.
```

```
  ?object rdf:type ?objecttype.
  ?objecttype fma:regional_part ?regionalparts
}


Result (extract) :


------------------------------------------------------------------------------------------------
| name                                        | objecttype   | regionalparts                      |
================================================================================================
| "BrainWeb-Moderate-Multiple-Sclerosis-with-Maps" | fma:Skull    | fma:Viscerocranium                 |
| "BrainWeb-Moderate-Multiple-Sclerosis-with-Maps" | fma:Skull    | fma:Neurocranium                   |
| "BrainWeb-Moderate-Multiple-Sclerosis-with-Maps" | fma:Skull    | fma:Calvaria                       |
| "BrainWeb-Moderate-Multiple-Sclerosis-with-Maps" | fma:Skull    | fma:Orbit                          |
| "IrcaDB1.1-patho-with-Luts"                 | fma:Kidney   | fma:Anterior_superior_renal_segment |
| "IrcaDB1.1-patho-with-Luts"                 | fma:Kidney   | fma:Inferior_renal_segment         |
| "IrcaDB1.1-patho-with-Luts"                 | fma:Kidney   | fma:Superior_renal_segment         |
| "IrcaDB1.1-patho-with-Luts"                 | fma:Kidney   | fma:Anterior_inferior_renal_segment |
| "IrcaDB1.1-patho-with-Luts"                 | fma:Kidney   | fma:Posterior_renal_segment        |
(...)
```

Figure 8. Example of a query using FMA knowledge; execution time was 18ms on a PC Intel i7 (Quad core, 8 GB of memory); the complete query answer as well as more sample queries are provided at http://vip.creatis.insa-lyon.fr/ontovip/modules/#queries

Simulation: Model annotations are exploited by a visualization tool helping the user defining simulation scenes. Models containing objects defined by meshes can be rendered in 3D and positioned w.r.t. imaging devices, for instance scanners or transducers. Annotations are exploited to enable the showing/hiding of specific model components in the simulation scene, and to change their color or opacity. This is illustrated Fig. 9 with the example of the ADAM model (heart-thorax model) [21].
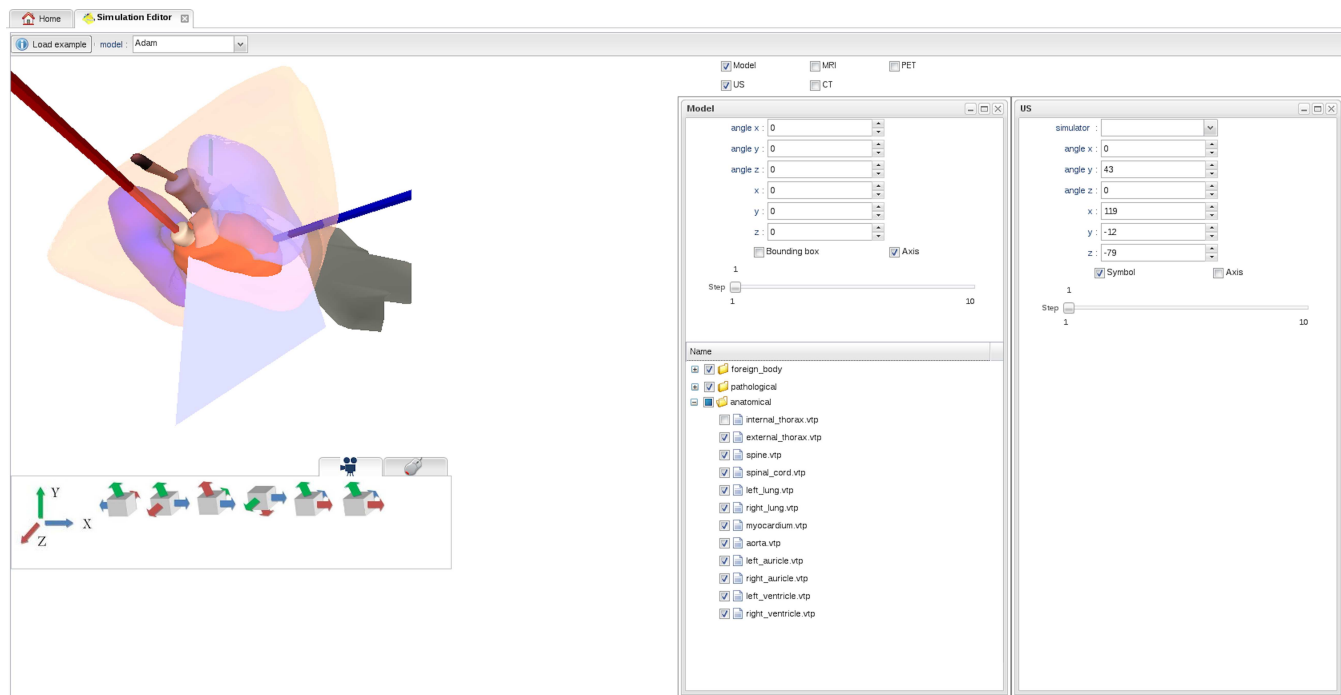
Figure 9. Screenshot of the visualization tool, where a simulation scene involving a thorax model and an ultrasonic transducer (grey object with sectorial imaging plane) was defined. Annotations are exploited to list the model components by layers, and allow their visual customization.

Image simulation can then be launched. Examples of images produced using the "BrainWeb-Severe-Multiple-Sclerosis-with-Maps" model (described above) are shown Fig. 10.
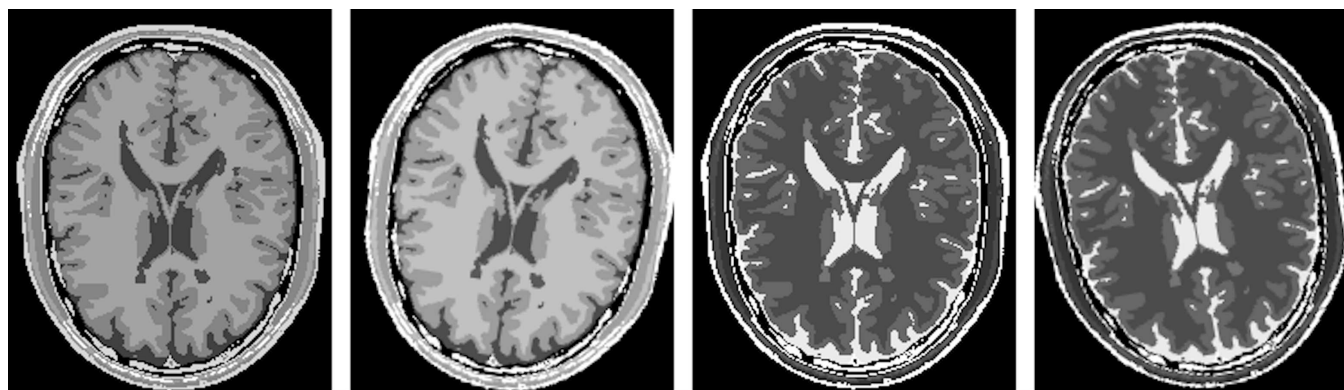

Figure 10. MRI simulations (3D) performed from the BrainWeb model with severe multiple-sclerosis lesions. Images were generated (SimuBloch V0.3 simulator) with different model positions w.r.t the scanner and different sequence parameters; from left to right: T1-weighted (TR=500ms, TE=8.4s) spin-echo with 2 different rotations w.r.t axis z, T2-weighted (TR=6530ms, TE=84ms) spin-echo with 2 different rotations w.r.t axis z.

4. Discussion

Domain covered and relevance to the medical image simulation domain's needs: As reminded earlier an ontology has two basic complementary functions, the first one is to provide a common vocabulary and the second one is to associate to this vocabulary formal knowledge represented in a logical language which in turn enables advanced querying and reasoning. The question whether a mere vocabulary would be sufficient for VIP is of course fully relevant. As far as interoperability with biological modeling software is concerned, one may consider that a common vocabulary for denoting shared information would probably be sufficient. However for the other motivations our assumption was that more complex characterization of entities were needed, for example for organizing anatomical structures in a part-whole hierarchy, or for relating objects to their physical properties.

The work presented in this paper is a first step in this direction. Especially, it does not aim at embracing the whole complexity of those biological models, but rather at focusing on the modest but still challenging objective of providing a common conceptualization for a large family of models used in medical image simulation, and gaining experience of their use in a model sharing platform such as the VIP platform. Extending this conceptualization to make it compatible with other kinds of models such as those produced in the @NeurIST [31] or euHeart [2] projects remains to be done. Nevertheless, we believe that the use of semantic web technologies and especially ontologies will facilitate the convergence and interoperation of these various models in the future and this was an important motivation for using them.

In this study we put emphasis on the reuse of existing ontologies and on the extraction of the relevant subsets that are needed for our application. Our concern in reusing ontologies like FMA, PATO, MPATH, RadLex and ChEBI is really to use ontologies that are likely to be chosen by biologists in their own model developments and thus to facilitate future interoperability with their models and software. However, such subsets are not easy to specify since simulation studies may involve a wide spectrum of objects, from physical phantoms to biological objects at various scales and concerning various species. Our strategy consisted in designing a flexible module extraction procedure to be able to easily extend those modules in the future if needed. An alternative could have been to include complete ontologies, but we felt that it was equally important to keep our application ontology as small as possible to enable the use of reasoners, supporting intelligent querying or reasoning, which would not be possible with large ontologies such as the entire FMA. One of the remaining problems is the consistency of the modules according to future evolutions of the original ontologies. Indeed, the extraction of the modules, even supported by automatic extraction tools, still needs manual operations (e.g. execution of the extraction query, importation in the global ontology, etc.).

Reuse of FMA: Our choice of FMA was motivated by the fact that FMA is a mature, highly recognized ontology. Moreover, it is quite exhaustive, at least for human anatomy. It also puts a strong emphasis on part-whole relationships, which is an important feature for us for the following reason. Anatomical object layer parts depict objects and refer to corresponding anatomical entities. It seemed important to us that users can specify anatomy at the level of precision they feel appropriate, e.g. `Head` as a whole, or distinguishing the different parts of the head, i.e. `Skin of head`, `Skull`, `Brain` etc. Since in our model an `anatomical object layer part` refers to exactly one `anatomical object` it is important that the knowledge embedded in the ontology actually represents the part-whole relationships that bridge these different levels of description. FMA provides this kind of knowledge, especially through the `fma:regionalPart` and `fma:constitutionalPart` object properties (and their inverse properties). However, we met several difficulties in extracting such properties from FMA. The

major one is that the OWL version of FMA that we used (FMA 3.1, available from the NCBO bioPortal) was in OWL Full, many entities being defined both as instances and as classes. A particular problem was that precisely the part-whole relationships were modeled as relationship between instances, whereas our intention was to model such knowledge as axioms attached to the classes, according to a regular DL model in which instances represent the anatomical structures instances referred to by each model. The origin of this problem lies in the fact that the original FMA ontology was represented in Protégé frames (the frame-based knowledge representation system developed in Stanford in the Mark Musen's group), a model in which an anatomical entity is represented both by a class and a metaclass, an elegant but rather complex way of managing selective inheritance of some attributes [32]. Translation into OWL DL of the native FMA ontology is a non trivial task and was clearly not feasible in the context of the VIP project [33]. Recent works such as [34] have improved solutions to this problem but were not available when the VIP project started. In practice, we limited ourselves at retrieving `fma:regionalPart` properties and associating them to the anatomical entities classes as annotation properties. This allowed to partly address the need of organizing anatomical structures in a part-whole hierarchy, as shown in section III-B. Further work would be needed to achieve the full retrieval of all part-whole relationships and the full use in query and reasoning of their intrinsic properties (such as transitivity).

Important limitations of the current conceptualization: Relation to time was addressed in a simplistic way, since our conceptualization considers discrete time samples only, and describes the physical properties of the imaged objects at those time points or instants. However, it does not characterize the state of the objects (e.g. systole or diastole), nor does it consider physiological processes (e.g. heart beating, breathing) and their abnormalities (e.g. heart arrhythmia, irregular breathing). This is certainly an interesting extension to consider in the future.

Choice of an extraction tool: We made the choice of vSPARQL as a tool to extract relevant subsets of existing ontologies. vSPARQL [16] (in conjunction with Gleen [17]) extends the SPARQL language and offers the possibility to create a recursive query. This feature is particularly useful to extract parts of an existing ontology without knowing its structure. Actually, few other solutions exist to perform such task. The simplest one is provided by the native CONSTRUCT clause of the SPARQL language (http://www.w3.org/TR/rdf-sparql-query/), and enhanced with the property paths expressions in SPARQL 1.1. The CONSTRUCT query returns a single RDF graph specified by a graph template. It allows specific triples to be extracted from an ontology in order to create a module. Alternatively, ModTool [35] is an alternate solution which offers tools to extract a module from existing ontologies. However, it provides less flexibility than vSPARQL in designing the query for the extraction. A review of the different approaches to create ontology modules is presented in [10].

Current status and limitations of the implementation: All the capabilities described above (section III.2) are available, and can be tested by connecting to the VIP Portal[6]. However, only the VIP project partners used the models' repository capabilities yet. Of course, only a subset of the capabilities of the ontology is exploited in the VIP implementation yet. For example, our ontological model offers a great flexibility to define pathological structures either using pathological entities (chosen from the MPATH extract) or defining them as anatomical structures (chosen from the FMA extract) bearing (using the `hasForQuality` object property) one or more pathological qualities (chosen from the PATO extract).

---

[6] http://vip.creatis.insa-lyon.fr/

5. Conclusion and perspectives

This work stresses very much the need of collaboration between biomodeling scientists and the medical imaging community. Such collaboration is essential to specify computer representations of biological objects and biological phenomena that are machine-processable, with explicit semantics. The basic technology supporting this is already available, thanks to XML-based syntaxes and ontology languages. However, the hardest aspect remains information semantics, which is hard to define and to share especially in application domains that require manipulating pluridisciplinary knowledge.

This project is exemplary of efforts made in this direction, with 1) a collaboration between simulation software scientists and image processing researchers to better understand future needs and uses of medical image simulation; 2) an effort to bridge the gap to the biologists' community, so that future models can really associate anatomy, physiology and physiopathology, and fulfill the needs of biomedical research and clinical applications.

CONFLICT OF INTEREST

The authors declare that they do not have any conflict of interest concerning the works reported in this paper.

REFERENCES

[1] P. Hunter, P. V. Coveney, B. de Bono, V. Diaz, J. Fenner, A. F. Frangi, et al. A vision and strategy for the virtual physiological human in 2010 and beyond. Phil. Trans. R. Soc. A, vol. 368, pp 2595-2614, 2010.

[2] D. Gianni, S. McKeever, T. Yu, R. Britten, H. Delingette, A. Frangi, et al. Sharing and reusing cardiovascular anatomical models over the Web: a step towards the implementation of the virtual physiological human project. Phil. Trans. R. Soc. A, vol. 368, pp 3039-3056, 2010.

[3] J. Tabary, S. Marache, S. Valette, W. Segars, and C. Lartizien. Realistic X-Ray CT simulation of the XCAT phantom with SINDBAD. In: IEEE NSS and MIC Conference, Orlando, USA, pp. 3980–3983, 2009.

[4] H. Benoit-Cattin, G. Collewet, B. Belaroussi, H. Saint-Jalmes, and C. Odet. The SIMRI project: a versatile and interactive MRI simulator. Journal of Magnetic Resonance Imaging, vol. 173, no. 1, pp. 97–115, 2005.

[5] B. Aubert-Broche, A. C. Evans, and L. Collins. A new improved version of the realistic digital brain phantom. Neuroimage, vol. 32, pp. 138-145, 2006.

[6] A. Reilhac, G. Batan, C. Michel, C. Grova, J. Tohka, D. L. Collins, et al. PET-SORTEO: Validation and Development of Database of Simulated PET Volumes. IEEE Transactions on Nuclear Science, vol. 52, pp. 1321–1328, 2005.

[7] S. Jan, D. Benoit, E. Becheva, T. Carlier, F. Cassol, P. Descourt, et al. Gate v6: a major enhancement of the gate simulation platform enabling modelling of ct and radiotherapy. Physics in medicine and biology, vol. 56, no. 4, pp. 881–901, 2011.

[8] J. Jensen and N. B. Svendsen. Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers. IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, vol. 39, no. 2, pp. 262–267, 1992.

[9] T. Glatard, C. Lartizien, B. Gibaud, B., R. Ferreira da Silva, G. Forestier, F. Cervenansky, et al. A Virtual Imaging Platform for multi-modality medical image simulation. IEEE Transactions on Medical Imaging, vol 32, no.1, pp. 110-118, 2013. doi: 10.1109/TMI.2012.2220154.

[10] J. Pathak, T. M. Johnson, and C. G. Chute. Survey of modular ontology techniques and their applications in the biomedical domain. Integr Comput Aided Eng., vol. 16, no. 3, pp. 225–242, 2009. doi:10.3233/ICA-2009-0315.2011.

[11] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, R. Oltramari, et al. The WonderWeb library of foundational ontologies and the DOLCE ontology. WonderWeb deliverable D18, final report (v. 1.0, 31-12-2003), 2003.

[12] L. Temal, M. Dojat, G. Kassel, and B. Gibaud. Towards an Ontology for Sharing Medical Images and Regions of Interest in Neuroimaging. Journal of Biomedical Informatics, vol. 41, no. 5, pp. 766-778, 2008.

[13] B. Gibaud, G. Kassel, M. Dojat, B. Batrancourt, F. Michel, A. Gaignard, et al. NeuroLOG: sharing neuroimaging data using an ontology-based federated approach. AMIA Annu Symp Proc., pp. 472-80, 2011.

[14] J. Y. Fortier, G. Kassel. Managing knowledge at the information level: an ontological approach. In: ECAI'2004 workshop on knowledge management and organizational memories. Barcelona (Spain), pp. 39-45, 2004.

[15] G. Kassel. A formal ontology of artefacts. Applied Ontology, 5 (3-4), pp. 223-246, 2010.

[16] M. Shaw, L. T. Detwiler, N. Noy, J. Brinkley, and D. Suciu. vSPARQL: A view definition language for the semantic web. Journal of Biomedical Informatics, vol. 44, no. 1, pp. 102-117, 2011.

[17] L. T. Detwiler, D. Suciu, and J. F. Brinkley. Regular Paths in SparQL: Querying the NCI Thesaurus. AMIA Annu Symp Proc, pp 161-165, 2008.

[18] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L.J. et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature Biotechnology, 25(11):1251-5, 2007.

[19] C. Rosse and J. L. V. Mejino. The foundational model of anatomy ontology. In: A. Burger, D. Davidson, and R. Baldock, editors, Anatomy Ontologies for Bioinformatics, volume 6 of Computational Biology, pp. 59–117, 2008.

[20] W. Segars, G. Sturgeon, S. Mendonca, J. Grimes, and B. M. W. Tsui. 4D XCAT phantom for multimodality imaging research. Med. Phys., vol. 37, no. 9, pp. 4902–4915, 2010.

[21] R. K. Kwan, A. C. Evans, and G. B. Pike. MRI simulation-based evaluation of image-processing and classification methods. IEEE Transactions on Medical Imaging, vol. 18, no. 11, pp. 1085–1097, 1999.

[22] R. Haddad. Un modèle numérique anthropomorphique et dynamique du thorax respirant et du coeur battant. Ph.D. dissertation, INSA Lyon, 2007.

[23] I. G. Zubal, C. R. Harrell, E. O. Smith, Z. Rattner, G. Gindi, and P. B. Hoffer. Computerized three-dimensional segmented human anatomy. Medical Physics, vol. 21, no. 2, pp. 299-302, 1994.

[24] A. Maedche, and S. Staab. Measuring Similarity between Ontologies. In: EKAW '02 Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, pp. 251-263, 2002.

[25] P. Schofield, G. Gkoutos, M. Gruenberger, J. Sundberg, and J. Hancock. Phenotype ontologies for mouse and man: bridging the semantic gap. Disease models & mechanisms, vol. 3, no. 5-6, pp. 281–289, 2010.

[26] G. V. Gkoutos, E. C. J. Green, A. M. Mallon, J. M. Hancock and D. Davidson. Using ontologies to describe mouse phenotypes. Genome Biology vol. 6:R8, 2004.

[27] C. P. Langlotz. RadLex: a new method for indexing online educational materials. RadioGraphics vol. 26, no. 6, pp. 1595-1597, 2006.

[28] P. de Matos, R. Alcántara, A. Dekker, M. Ennis, J. Hastings, K. Haug, et al. Chemical entities of biological interest: an update. Nucleic Acids Research, vol. 38, D249-54, 2010.

[29] G. Kassel. Integration of the DOLCE top-level ontology into the OntoSpec methodology. LaRIA Research Report 2005–2008. Available from: http://hal.ccsd.cnrs.fr/ccsd-00012203. 2005.

[30] D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes et al. Design and construction of a realistic digital brain phantom. IEEE Transactions on Medical Imaging, vol.17, No.3, pp. 463-468, 1998.

[31] D12.3 Deliverable: Global Schema of the Disease (WP 4), @neurIST project: Integrated biomedical informatics for the management of cerebral aneurysms, 2009.

[32] N. F. Noy, M. A. Musen MA, J. L. V. Mejino Jr, and C. Rosse. Pushing the envelope: challenges in a frame-based representation of human anatomy. Data and Knowledge Engineering, 48(3):335–59, 2004.

[33] O. Dameron, D. L. Rubin, and M. A. Musen. Challenges in converting frame based ontology into OWL: the Foundational Model of Anatomy case-study. In: AMIA annual symposium:Washington DC, pp. 181–5, 2005.

[34] C. Golbreich, J. Grosjean, and S. J. Darmoni. The Foundational Model of Anatomy in OWL 2 and its use. Artificial Intelligence in Medicine, vol 57, pp. 119-132, 2013.

[35] P. Doran, V. Tamma, and L. Iannone. Ontology module extraction for ontology reuse: an ontology engineering perspective. In: Proceeding CIKM '07 Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, pp. 61-70, 2007.

Appendix A

This Appendix contains an example of vSPARQL [16] query designed to create a module in OntoVIP. We use as illustration the anatomical objects extracted from the FMA. A list of terms (e.g. brain, heart), present in existing simulation models, was used to determine the concepts to extract from FMA (e.g. `fma:Brain`, `fma:Heart`). Once FMA concepts identified, we designed a recursive query allowing extracting the concepts along with their information (e.g. FMAID, label, etc.) and their cotopy (i.e. all the concepts from the targeted concept to the root of the ontology). We also used Gleen [17] in the query to extract the regional parts of the targeted concepts.

The query below extracts the concept `fma:Brain`, its information, its cotopy and its regional parts. The full extraction query for all anatomical objects contains the union of multiple queries where `fma:Brain` is replaced by each of the targeted concepts (e.g. `fma:Heart`, `fma:Lung`, etc.). The construction of this full query is automatic and takes as input a list of FMA concepts. Thus, concepts can be easily added/removed from this list (for example from user's feedback) and a new extraction can be performed. More details and descriptions of extraction procedures for the other modules of OntoVIP are provided on this webpage: http://vip.creatis.insa-lyon.fr/ontovip/modules/

```
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX fma:<http://sig.uw.edu/fma#>
PREFIX gleen:<java:edu.washington.sig.gleen.>

CONSTRUCT {?x ?y ?z.}
FROM NAMED <extracted_ontology> [
CONSTRUCT {?a ?b ?c. ?c rdfs:subClassOf ?w. ?a rdfs:subClassOf ?t. ?c fma:FMAID ?fmaid . ?c
fma:definition ?definition . ?c rdfs:comment ?comment. ?c rdfs:label ?label. ?a fma:FMAID ?fmaid2 . ?a
fma:definition ?definition2 . ?a rdfs:comment ?comment2. ?a rdfs:label ?label2. ?a fma:name
?freesurfer_name. ?a fma:Freesurfer_ID ?freesurfer_id.}
FROM <http://sig.uw.edu/fma>
WHERE { (fma:Brain '([fma:regional_part]|[fma:regional_part_of])' ?f) gleen:Subgraph (?a ?b ?c) . ?c
rdfs:subClassOf ?w. ?a rdfs:subClassOf ?t. FILTER(!isBlank(?w)). FILTER(!isBlank(?t)). ?c fma:FMAID
?fmaid .  OPTIONAL{?c fma:definition ?definition}. OPTIONAL {?c rdfs:comment ?comment }. ?c rdfs:label
?label. ?a fma:FMAID ?fmaid2 .  OPTIONAL{?a fma:definition ?definition2}. OPTIONAL {?a rdfs:comment
?comment2 }. ?a rdfs:label ?label2. OPTIONAL{?a fma:Freesurfer ?obj. ?obj fma:name ?freesurfer_name.
?obj fma:Freesurfer_ID ?freesurfer_id}.}
UNION
CONSTRUCT {?w rdfs:subClassOf ?next. ?w fma:FMAID ?fmaid . ?w fma:definition ?definition . ?w
rdfs:comment ?comment. ?w rdfs:label ?label. ?w fma:name ?freesurfer_name2. ?w fma:Freesurfer_ID
?freesurfer_id2.}
FROM <http://sig.uw.edu/fma>
FROM NAMED <extracted_ontology>
WHERE {
GRAPH <extracted_ontology> {?next2 rdfs:subClassOf ?w.} .
?w rdfs:subClassOf ?next. ?w fma:FMAID ?fmaid .  OPTIONAL{?w fma:definition ?definition}. OPTIONAL {?w
rdfs:comment ?comment }. ?w rdfs:label ?label. OPTIONAL{?w fma:Freesurfer ?obj2. ?obj2 fma:name
?freesurfer_name2. ?obj2 fma:Freesurfer_ID ?freesurfer_id2}.
FILTER(!isBlank(?next)).
}
UNION
CONSTRUCT {?t rdfs:subClassOf ?next. ?t fma:FMAID ?fmaid . ?t fma:definition ?definition . ?t
rdfs:comment ?comment. ?t rdfs:label ?label. ?t fma:name ?freesurfer_name3. ?t fma:Freesurfer_ID
?freesurfer_id3.}
FROM <http://sig.uw.edu/fma>
FROM NAMED <extracted_ontology>
WHERE {
GRAPH <extracted_ontology> {?next2 rdfs:subClassOf ?t.} .
?t rdfs:subClassOf ?next. ?t fma:FMAID ?fmaid .  OPTIONAL{?t fma:definition ?definition}. OPTIONAL {?t
rdfs:comment ?comment }. ?t rdfs:label ?label. OPTIONAL{?t fma:Freesurfer ?obj3. ?obj3 fma:name
?freesurfer_name3. ?obj3 fma:Freesurfer_ID ?freesurfer_id3}.
FILTER(!isBlank(?next)).
}
UNION
CONSTRUCT {fma:Brain rdfs:subClassOf ?super .fma:Brain fma:FMAID ?fmaid . fma:Brain fma:definition
?definition . fma:Brain rdfs:comment ?comment . fma:Brain rdfs:label ?label .}
FROM <http://sig.uw.edu/fma>
WHERE {fma:Brain rdfs:subClassOf ?super .fma:Brain fma:FMAID ?fmaid .  OPTIONAL{fma:Brain fma:definition
?definition} . OPTIONAL{fma:Brain rdfs:comment ?comment } .fma:Brain rdfs:label ?label .
FILTER(!isBlank(?super)).}
UNION
CONSTRUCT {?super rdfs:subClassOf ?next . ?super fma:FMAID ?fmaid . ?super fma:definition ?definition .
```

```
?super rdfs:comment ?comment. ?super rdfs:label ?label}
FROM <http://sig.uw.edu/fma>
FROM NAMED <extracted_ontology>
WHERE {   GRAPH <extracted_ontology> {?next2 rdfs:subClassOf ?super} .
?super rdfs:subClassOf ?next . ?super fma:FMAID ?fmaid .  OPTIONAL{?super fma:definition ?definition}.
OPTIONAL {?super rdfs:comment ?comment }. ?super rdfs:label ?label
FILTER(!isBlank(?next)).
}
] WHERE {GRAPH <extracted_ontology> {?x ?y ?z }.}
```