# AIM@SHAPE

Advanced and Innovative Models And Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects

**IST NoE No 506766**

## Deliverable D1.2.1.1

# Ontology for Virtual Humans 1st version


**Circulation:** [1]CO

**Partner(s):** **EPFL, UNIGE, ITI, IMATI, MPII**



**Authors:** A. Garcia-Rojas, L. Moccozet, G. Vasilakis, M. Pitikakis, M. Spagnuolo, A. Belyaev



**Version:** 01

**Stage:** 100% Draft

**Date:** Tuesday, December 20, 2005


[1] Please indicate the dissemination level using one of the following codes:

**PU** = Public

**PP** = Restricted to other programme participants (including the Commission Services).

**RE** = Restricted to a group specified by the consortium (including the Commission Services).

**CO** = Confidential, only for members of the consortium (including the Commission Services).

Copyright
© Copyright 2005 The AIM@SHAPE Consortium

consisting of:

| | |
|---|---|
| CNR-**IMATI**-GE | C.N.R. – Istituto di Matematica Applicata e Tecnologie Informatiche Dept. of Genova, Italy |
| **DISI** | Università di Genova – Dipartimento di Informatica e Scienze dell'Informazione, Italy |
| **EPFL** | École Polytechnique Federale de Lausanne, Switzerland |
| FhG/**IGD** | Fraunhofer Institut für Graphische Datenverarbeitung, Germany |
| **INPG** | Institut National Polytechnique de Grenoble, France |
| **INRIA** | Institut National de Recherche en Informatique et Automatique, France |
| **ITI**-CERTH | Informatics and Telematics Institut – Center for Research and Technology Hellas, Greece |
| **UNIGE** | Université de Genève, Switzerland |
| **MPII** | Max-Planck-Institut für Informatik, Germany |
| **SINTEF** | Stiftelsen for industriell og teknisk forskning ved Norges Tekniske Høgskole, Norway |
| **TECHNION** | Technion – Israel Institute of Technology, Israel |
| **TUD** | Technische Universität Darmstadt, Germany |
| **UU** | Utrecht University, Netherlands |
| **WEIZMANN** | Weizmann Institute of Science, Israel |

This document may not be copied, reproduced, or modified in whole or in part for any purpose without written permission from the AIM@SHAPE Consortium. In addition to such written permission to copy, reproduce, or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.

All rights reserved.

This document may change without notice.

## Document History

| Vers. | Issue Date | Stage | Content and changes |
|---|---|---|---|
| 1 | 22 November 2005 | 100 % | First version of the ontology |
| 2 | | | |

## Executive Summary

This document contains a description of the deliverable **D1.2.1.1** of the IST NoE AIM@SHAPE. The deliverable ***D1.2.1.1 – Ontology for Virtual Humans 1st version*** – is intended to provide a first version of the ontology for Virtual Humans, which is part of the ontology development process in the network.

The task leader is **EPFL** and is actively supported by all other AIM@SHAPE partners.

The document is structured as follows. Section 1 gives a short introduction and outlines the motivation behind developing an ontology for virtual humans. Section 2 describes issues regarding the knowledge of domain of creating virtual humans: Modeling, Animation and Interaction. Section 3 describes the process of developing the ontology with the chosen specification language (OWL). Also in Section 3 the purpose of the Virtual Human Ontology is described and several usage scenarios are identified and decomposed into simple Competency Questions, specifying necessary functionality and measuring the completeness of the ontology.

# Table of Contents

**1 INTRODUCTION** .......... **5**

**2 VIRTUAL HUMANS** .......... **6**

2.1 VIRTUAL HUMAN MODELLING .......... 6
*2.1.1 Landmarks* .......... *6*
*2.1.2 Garments and Accessories* .......... *7*
2.2 VIRTUAL HUMAN ANIMATION .......... 8
*2.2.1 Skeletal Structure* .......... *8*
*2.2.2 Animation Control* .......... *9*
2.3 VIRTUAL HUMAN INTERACTIONS .......... 10

**3 VIRTUAL HUMAN ONTOLOGY COMPONENTS** .......... **12**

3.1 PURPOSE OF THE ONTOLOGY .......... 12
3.2 COMPETENCY QUESTIONS (CQS) .......... 12
3.3 ONTOLOGY DESIGN .......... 14
*3.3.1 Classes for Virtual Human Modelling* .......... *16*
*3.3.2 Classes for Virtual Human Animation* .......... *16*
*3.3.3 Classes for Virtual Human interactions* .......... *20*
3.4 USAGE SCENARIOS .......... 21

**4 REFERENCES** .......... **23**

# 1 INTRODUCTION

Ontology development is a continuous process and the results we present here constitute the first version of this work in progress. Nevertheless, the current ontology provides a good starting point towards the creation of a more versatile, extensible and reusable representation of Virtual Humans.

There has been some work done and presented regarding the ontology of Virtual Humans in [VHOnto]. This document is based on this presentation but also includes some evolution. Moreover, in this document the ontology developed is described in much more detail.

An ontology representation of a virtual human must be closely linked to the associated geometrical one. It is particularly required to be able to move from the geometric representation towards the ontology-driven -semantic- one: with the analysis of the 3D graphical representation in order to query the 3D models for semantic information. It is also required to be able to go from the ontology description to the graphical representation: with the integration of the semantic descriptors in the modeling and animation process, which means that we need to be able to construct the graphical representation of a virtual human from the semantic descriptors.

This is made possible thanks to an intermediate layer of human shape descriptors (features, landmarks, segments, etc.). Nowadays, many detailed 3D datasets of human bodies are available, and with the current scanning technology, new ones are relatively easy to produce. As a result, recent modeling approaches, based on real data and statistical analysis of shape databases, should allow for controlling the synthesis of human shapes with high-level body descriptors based on landmarks and features.

In the next section we present the knowledge of the domain of the Virtual humans. This comprehend form its geometric creation until its population inside the virtual environment. As it has already been mentioned, the ontology is still work in progress, but it already includes the essential considerations for the creation of virtual human objects.

# 2 VIRTUAL HUMANS

Virtual Humans, as 3D graphical representations of human beings have a large variety of applications. Within inhabited Virtual Environments, Virtual Humans (VHs) are a key technology that can provide virtual presenters, virtual guides, virtual actors, and be used to show how humans behave in various situations.

Creating Virtual Humans is a complex and time consuming task which involves several areas in Computer Science: Artificial Intelligence, Computer Graphics, Geometric Modeling, Multimodal Interfaces, etc. Our main contribution focuses on proposing a semantics-based method for organizing the various types of data that constitute a Virtual Human. The knowledge related to the synthesis, animation and functionalities of VHs is formally specified in the form of the proposed ontology.

## 2.1 Virtual Human Modelling

Body shape reconstruction and synthesis as described by T. Dey in [Dey05] mentioned that recent advances in scanning technology [Daanen98] let to rapidly emerge what is called “sample based geometric modeling” for digital modeling of physical objects from sample points. The basic idea of these modeling methods relies on the usage of acquisition devices such as 3D scanners for extracting geometric data from a real instance. Because the source of the models is real data, they are suitable for producing realistic looking objects. However, acquisition devices do not provide “ready-to-use” results and post-processes are required in order to obtain an accurate shape. Acquired data are usually noisy, over-sampled and incomplete. The integration of pre-existing knowledge in the automatic reconstruction process should greatly improve the accuracy of the resulting instance to capture all the features of the object. Automating this approach requires the extraction of high-level information based on morphological landmarks in order to apply knowledge-based reconstruction methods.

The reconstruction scenario we are depicting starts with a real data acquisition process; then, the acquired model undergoes post-process reconstruction phases to update real data and complete the animation control structures; last, accessories and objects are specified and attached to the virtual human object. At all these stages, the ability to extract and attach semantic information from and to human body shapes is crucial. This mainly results in decomposing the shape into meaningful segmented parts or in locating anthropometric landmarks over the body model.

### 2.1.1 Landmarks

The most common features involved in human shape synthesis are landmarks. Landmarks and segments provide low-level semantic descriptors from which it is possible to derive higher-level ones. Landmarks are points of correspondence on each object of the same kind that match between and within populations [BookStein97]. The widely adopted landmarks structure for human shape is the one proposed in the H-ANIM standard [h-anim] description, as shown in figure 1 (top).

The computational methods involved in the extraction of features such as landmarks or shape segmentation must comply with the following constraints:

• Landmarks extraction and morphological segmentation results must be anthropometrically consistent. Extracted features and segmentation must be associated to anthropometric features and segments.

• Landmarks extraction and morphological segmentation results must be consistent and almost invariant from one data set to another.

In figure 1, we show some examples of landmarks extraction based on a multi-scale morphological analysis of the human shape. These features are extracted with a tool called Tailor, which is based on a multi-scale morphological analysis method [Mortara03]. This method decomposes the surface into meaningful shape features, like tips, tubular protrusions, concave regions, sharp points, etc.

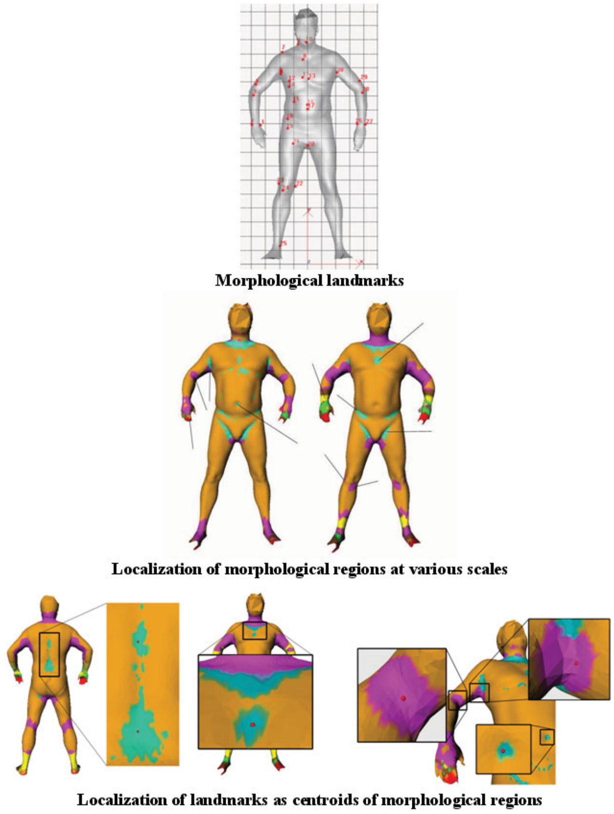


**Figure 1 Landmarks structure for human shape**

The Landmarks are useful for a variety of proposes. They can be used for attaching clothing and jewelry, and can be used as end-effectors for inverse kinematics applications. They can also be used to define eye-points and viewpoint locations.

### 2.1.2 Garments and Accessories

Garments and accessories are objects such as clothes, jewels, hats, glasses, etc., that are attached to the human shape. Their motion and animation depends on the motions and animations of the human shape itself. Attaching accessories to a human shape involves locating where they should be placed on the body shape and extracting measurements information in order to fit the accessories to the body shape. The figure 2 shows a garment positioning example.

A garment is composed by some cloth patterns sewed with each other. A piece of clothing corresponds to a part or segment of the human body, for example the sleeve corresponds to an arm. And to put the cloth in the right position we use the landmarks identified in the corresponding human segment.

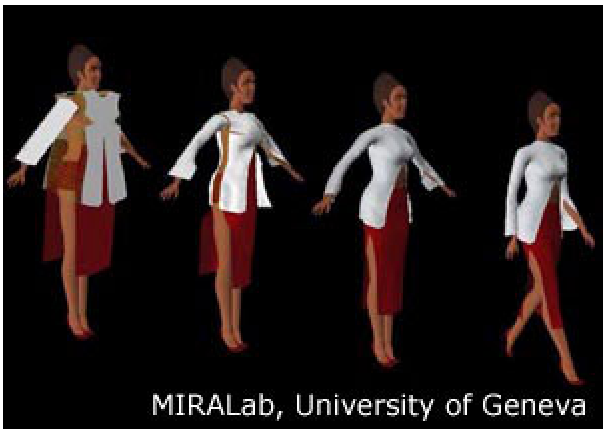


**Figure 2 Garment positioning**

At first glance, correctly placing an accessory intuitively relies on a morphological segmentation of the human shape, e.g. a watch will be placed at the border between the hand and the forearm; a shirt will cover the trunk and the arms. Morphological segmentation of the human shape can also be used to optimize the simulation process when flexible accessories such as clothes are simulated according to the human shape animation.

## 2.2 Virtual Human animation

The human body shape is a typical example family of an articulated physical object: it does not have only one shape but many, corresponding to all the possible postures that the underlying articulated skeleton can take.

### 2.2.1 Skeletal Structure

When acquiring scan data, we only obtain a single static snapshot of the body shape. As such and for many range applications, this static snapshot is not sufficient as it does not capture all the possible degrees of flexibility of the human shape. To mimic the flexible and dynamic behavior of the human shape, the traditional approach uses skeleton-driven deformations, a classical method for the basic skin deformation that is among the most widely used techniques in 3D character animation. Body animation binds a 3D shape to an articulated control skeleton. Binding information is then used to deform the body shape according to control skeleton motion.

This skeletal structure has been standardized by the H-Anim [h-anim] specification which considers that the human body consists of a number of *segments* (such as the forearm, hand and foot) which are connected to each other by *joints* (such as the elbow, wrist and ankle). An H-Anim file contains a set of Joint nodes that are arranged to form a hierarchy. Each Joint node can contain other Joint nodes, and may also contain a Segment node which describes the body part associated with that joint. Each Segment can also have a number of Site nodes or Landmarks, which define locations relative to the segment. This standard gives support to the creation of interchangeable humanoids, behaviors and animations.

**Figure 3 Human shape associated with a skeletal structure**

### 2.2.2 Animation Control

The main computer animation methods to animate characters are: Motion capture animation, Key-frame animation, parametric interpolation and procedural or algorithmic animation. Motion capture animation measures and records action performed by an actor for immediate or delayed analysis and replication. It requires matching the virtual character's skeleton with measured movements of the MoCap's sensors in the body of the actor. The motion capture of the human walking applied to a synthetic actor gives excellent results, because the data comes directly from the real world. However, if the animation needs to be modified, the sequence must be recaptured.

In Key-frame animation the animator specifies the important key frames by defining the exact posture at a defined moment in the time, and the computer generates the in-betweens automatically using interpolation.

Animation by parametric interpolation is characterized by parameters with defined values over time. The computer calculates the intermediate values of the parameters by interpolation and recalculates the scene with the interpolated values.

Finally the procedural or algorithmic animation is based on the laws of physics. The advantage of this kind of animation provides high level of control but it requires a lot of mental work and some times it has low level of realism.

These methods of animation provide low level of motion systems and they do not provide autonomous behavior. For making a VH "to do something" is represented in levels of animation: Interaction with the environment, task level animation and behavioral animation and autonomous actors.

Therefore, VH creation also aims to provide virtual characters with realistic behavior, which implies endowing them with autonomy in an inhabited virtual environment (VE). A realistic Virtual Human behavior lies in a good simulation of its autonomy. This autonomy consists in self awareness (sensorial behavior) and task performance; but also in interaction with users, other virtual humans or with other 3D elements in the virtual environment.

The implementation of behavioral controllers is used for creating VH behavioral animation. They aim to control the VH behavior in a believable way. But they also implement individualization for each VH, otherwise VHs would act more like virtual robots. The incorporation of human characteristics (as gender, age, personality, emotions, etc.) is used to impart this individuality.

Body movements in each person are defined by the body's morphological characteristics. Those characteristics relevant in human movements are: gender, height, weight and age, which are inherent to the character's shape. Body shape morphology is important to be able to adapt motions to different characters as: avoid self body collisions, appropriate gender posture, changing velocity according to age, etc.

For the case of personality and emotions characteristics, they are implemented using psychological models. The computational model commonly used to describe personality is the Five Factor model which uses five dimensions: Extraversion, Agreeableness, Conscientiousness, Neuroticism, and Openness. Although there are other models considered depending on the usage. Jung's Model describes the differences of personality using categories of introversion and extroversion and also using cognitive and emotional functions. Or the Millon Model considers that personality can be viewed as a result of twelve bipolar attributes: openness versus preservation, modification versus accommodation and individualism versus protection.

To describe emotion there are several models for different objectives, e.g. Eckman's for face expression, OCC for cognition implementation, activation-evaluation for speech, etc. These models are readily amenable to the intentional stance, and so ideally suited to the task of creating concrete representations of personality and emotions with which to enhance the illusion of believability in computer characters.

Synthesizing autonomous behavior comprehend various techniques that provide the character with the ability of performing tasks, communicating, socializing, taking decisions, etc. by itself. In this level animations involve many other scientific fields of knowledge such as artificial intelligence, psychology, biology, etc. Therefore for this first version of the ontology we limit ourselves to consider that the human body can be animated through its articulations, the existence of animation sequences and behavioral controllers that drive its movements which use psychological models for imparting VH individuality in the behavior.

## 2.3 Virtual Human interactions

Once a virtual human is immersed inside a virtual environment, it has to be able to act as its real counterpart. Therefore it cannot be limited to a soup of triangles without higher level information. This is particularly true for virtual humans. Whenever they are included within a virtual environment, they are expected to move and interact with this environment. Obviously, the 3D body shape and even the control animation structure do not include the required level of information for modeling an "active" human shape.

The necessity to model interactions between an object and a virtual human appears in most applications of computer animation and simulation. Such applications encompass several domains, as for example: virtual autonomous agents living and working in virtual environments, human factors analysis, training, education, virtual prototyping, and simulation-based design. Commonly, simulation systems perform agent-object interactions for specific tasks. Such an approach is simple and direct, but most of the time, the core of the system needs to be updated whenever one needs to consider another class of objects. *Smart Objects* are an interesting way to model general agent-object interactions based on objects containing interaction information of various kinds: intrinsic object properties, information on how to interact with it, object behaviors, and also expected agent behaviors.

The smart object approach [Kallmann99] extends the idea of having a database of interaction information. For each object modeled, we include the functionality of its moving parts and detailed commands describing each desired interaction, by means of a dedicated script language. A feature modeling approach is used to include all desired information in objects. In the Figure 4 there is an example of the interaction of a virtual human and a coffee machine, it shows the proper manner of holding the elements of the machine.

**Figure 4 Interaction of Virtual Human and Smart Object**

# 3 VIRTUAL HUMAN ONTOLOGY COMPONENTS

An ontology is defined to be a formal specification of a shared conceptualization [Gruber93]. Virtual Humans are complex entities composed by well defined features, and functionalities. Concepts and techniques related to the creation and exploitation of VHs such as those described in previous sections are shared by the research community. Our effort is targeted at unifying such concepts and representing them in a formal way. A formal representation refers to the fact that VH representations and their associated semantics shall be both human and machine readable. The Web Ontology Language (OWL) is a specification used to represent the meaning of concepts and the relations between them. Among the available ontology languages OWL has recently become the adopted standard, offering the richest vocabulary in order to model the information that is inherent in a domain of knowledge.

Our objective is to support the creation and reuse of VHs by exploiting their underlying semantics. Associating semantic information to the components of a virtual environment has proved to be useful in terms of component reuse, content adaptation, etc. In [Gtz_IJCAT05], Gutiérrez et al. defined an object representation based on the semantics and functionality of interactive digital items – virtual objects – within a Virtual Environment (VE). Every object participating in a VE application is considered as a dynamic entity with multiple visual representations and functionalities. This approach allows for dynamically scaling and adapting the object's geometry and functions to different scenarios. In [Gtz_SVE05], the semantic model presented in [Gtz_IJCAT05] was complemented with an ontology of objects that allowed for expressing the relationships between interaction devices and virtual entities in a VE. The present work builds upon the acquired experience and focuses on a single type of virtual entity: Virtual Humans.

We have defined this first version of the VH ontology based on the competency questions listed in section 3.2.

## 3.1 Purpose of the ontology

The Virtual Humans (VH) ontology aims at organizing the knowledge and data of three main research topics and applications involving the virtual representations of humans:

- Human body modeling and analysis: morphological analysis, measuring similarity, model editing and reconstruction.
- Animation of virtual humans: autonomous or pre-set animation of VH.
- Interaction of virtual humans with virtual objects: virtual –smart– objects that contain the semantic information indicating how interactions between virtual humans and objects are to be carried out.

## 3.2 Competency questions (CQs)

The development of an ontology usually starts by defining its domain and scope. That is, by defining several basic questions (competency questions) that constitute requirements regarding the expressiveness and the functionality of the ontology. Competency questions (CQs) are one of the best ways to determine the scope of the ontology. CQs consist of a list of questions that a knowledge base based on the ontology should be able to answer [Gruninger et al.]. The proposed ontology should be able to answer the following categories of competency questions:

**Model history**

Is this model obtained by editing another model?
What features have been changed on model X?
What tools where involved in the synthesis/modification of this VH?
Who performed the task T on the model X?

**Features listing**

What is the height of the model?
Is the model male or female?
Is the model has cultural identification?
What are the features of this model?
Is this model obtained artificially or it represents a real person?
Which VH have a landmark description?
Which are the available structural descriptors for a particular VH?
Which aspects of the shape are described by the structural descriptor related to a particular VH?
Which are the standing (seating, walking, etc.) VH?
How is the body model represented? (a mesh, a point set, etc.)
Is the VH complete (does it have a skeleton, a hierarchy of body parts, a set of landmarks attached to it?)?

**Questions whose answer is a function of low/high level features**

For most of the answers to these questions the ontology needs more classification definitions, but still the answers can be provided with the information here defined.
Which are the VH that are fat/slim/short?
Is this VH a child or an adult?
Does it have a long nose?
Does it miss any body part?
Does this VH match another VH (or how much do they match)? In particular: are they in the same posture?
Do they have the same structure?
Do they have similar parts (same arm length, same fatness, similar nose?)?
Do they have similar anthropomorphic measures (in terms of landmarks)?
Is the model suitable for animation?
How will this VH look like after 20 years?
With 20 kg more? With another nose?
Does this model fit this cloth?
What VH do I get if I put the head of VH1 on the body of VH2?

**Animation sequences**

What model does this animation use?
What are the joints affected by this animation sequence?
Are there any animation sequences lasting more than 1 minute that are suitable for this VH?
Are there any running/football playing animation sequences for this kind of VH?
Can the animation sequence X be applied to the VH Y (in the case of key-frames for skeleton-based animation this would basically depend on the possibility to match the key-frame data to the skeleton of the VH)?

**Animation algorithms**

What are the input and output channels of a particular Behavior controller (animation algorithm)?
What are the models suitable to be animated with this algorithm?
Does this VH have a vision sensor attached?
Can this VH react to sound events in its virtual environment?

**Interaction with objects**

What capabilities does an object provide?
What are the actions the human can execute on the object?
What are the characteristics of an object (structure, physical properties, etc.)?
How can the object be grasped?

## 3.3 Ontology design

The technology used for the implementation of the ontology is OWL, a formal language that precisely specifies the semantic relationships among entities. This language offers more ways of entity relation, cardinality relation between entities and other logical operations when compared with older ontology languages like RDF Schema. The definition of the ontology was made using Protégé [Protege05] which is an open source ontology editor and knowledge-base framework.

The starting point of the ontology is the Virtual Human class. Later on in this section we will explain the development of the three knowledge areas involving the Virtual Human creation: modeling, animating and interaction with Virtual Humans.

The Virtual Human class is a full-body or partial representation of a human being. A Virtual human can be based on another virtual human or a virtual human can be used to generate a new one. The model can be synthesized in a variety of ways and can represent a real or a virtual person.

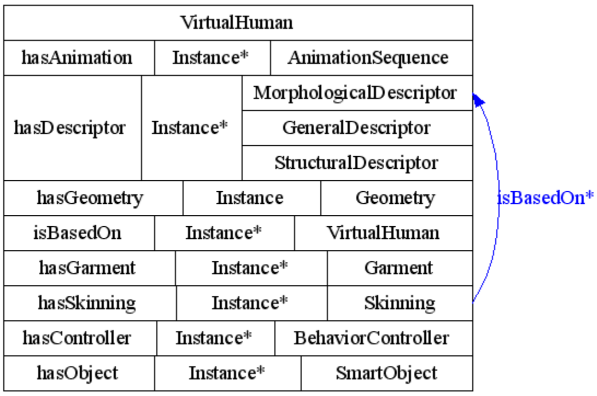


**Figure 5 Virtual Human class**

A simple Competency Question capturing some of the functionality of the Virtual Human class shown in Figure 5 is: *find models that have been predecessors of other models.* The CQ can also be modified to be more specific and more useful: *Is this model based on another model?*

The latter question can be formulated in a pseudo query language as follows:

(retrieve (isBasedOn *this-model* ?model)) [2]

The query will retrieve all the Virtual Human models that this model is based on. This does not involve only direct relations. The *isBasedOn* relation has been defined in the ontology to be a *transitive* property. For the question above this means that if *this-model* is based on model X, and model X is based on model Y, then model Y will be retrieved as well. Here we simply take advantage of the expressiveness of OWL which clearly surpasses that of older ontology languages. The key in this example is OWL's support for transitive properties.

[2] In the pseudo query language literal values are shown in italics and variables start with the character "?"

The next level to describe a VH is defined with descriptor classes. We have created the abstract class *Descriptor* which is further specialized in several sub-classes that contain the main features needed for describing a VH. These sub-classes are General Descriptor, Structural Descriptor and Morphological Descriptor and are shown in Figure 6.

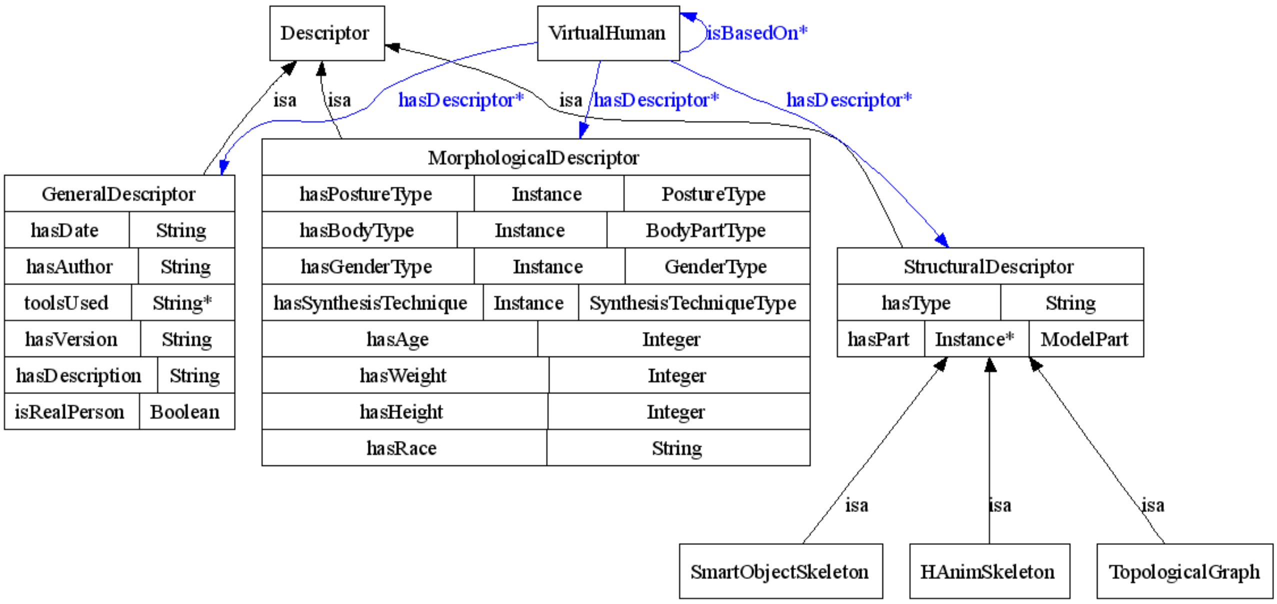


**Figure 6 Virtual Human Descriptors**

The General descriptor contains information of the process of creation of the model, like: who created the model, when, with which software, etc. This class together with the property *isBasedOn* of the VH class can help us to have the history of the model and map its evolution. As an example we have the CQ:

*What tools were involved in the synthesis/modification of this VH?*

This can be formulated in the pseudo query language as

(retrieve (isBasedOn *this-model* ?model) (toolsUsed ?tool ?model))

and will retrieve all tools used in the creation of this VH object, as well as to the creation of all VH objects on which it is based.

VHs are characterized by a set of attributes named Morphological descriptors that are inherent to the human shape (sex, approx. age, race, etc.); and also other information useful for designer as synthesis technique which says whether the model was made from scanning or from scratch with a 3D tool, etc. Some CQs that this class will help to answer are:

*What is the height of the model?*, which can be formulated as:
(retrieve (hasHeight *this-model* ?height))
*Is the model male or female?*, which can be formulated as:
(retrieve (hasGenderType *this-model* ?gender))

The structural descriptor defines the animation oriented structures such: H-Anim and Smart Object skeletons; and others such as topological graphs. These structures are subclasses of the Structural Descriptor. Some related CQs include:

*Which are the available structural descriptors for a particular VH?*
(retrieve (hasDescriptor *this-model* ?descriptor) (?descriptor *StructuralDescriptor*))
*Does the VH have an h-anim structure?*
(retrieve (hasDescriptor *this-model* ?descriptor) (?descriptor *HAnimSkeleton*))

These kinds of questions are of interest to animators, because some types of animation files are only applicable to VHs with a specific skeletal structure.

### 3.3.1 Classes for Virtual Human Modelling

The shape representation of the virtual humans is placed in the abstract class Geometry that contains a general description of 3D Shapes used to represent a VH body or parts of it. A set of landmarks can be associated to any geometry as well as textures and other generic information. These Landmarks are a placeholder to store information associated to a particular location in a 3D geometry. Landmarks can be used for attaching clothing and jewelry, as end-effectors for inverse kinematics applications or to define eye points and viewpoint locations.

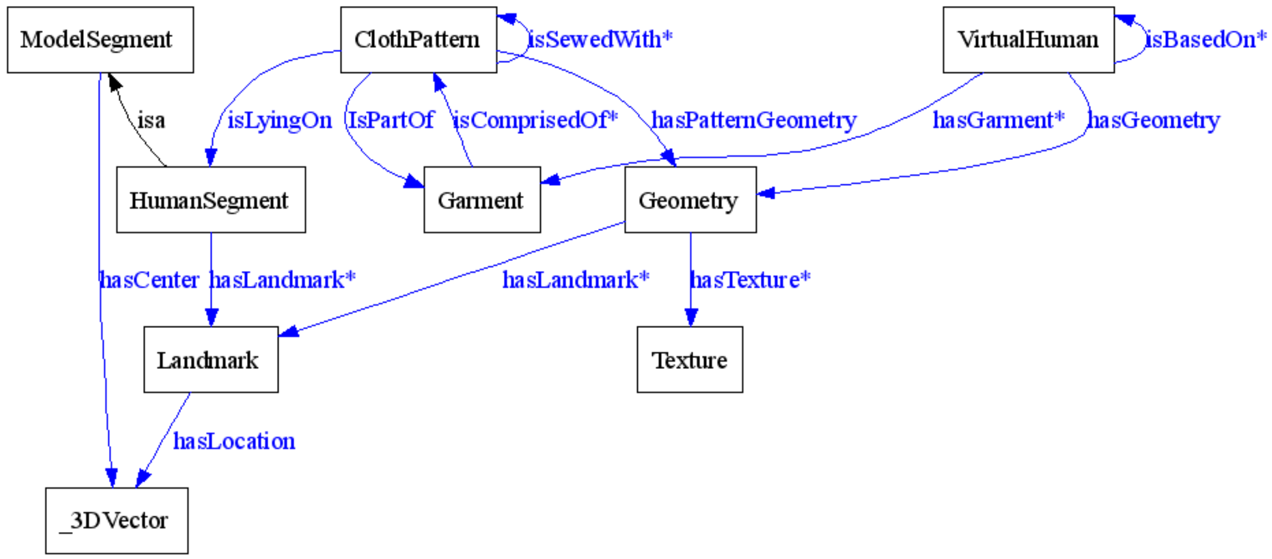


**Figure 7 Virtual Human modelling**

Here we might be interested in finding a VH with landmarks:
*Which VH have a landmark description?* We can formulate the question as:
(retrieve (hasGeometry *this-model* ?model) (hasLandmark ?model ?landmark))

To include the garments in our ontology we are based on the ontology for virtual garments presented in [Fuhrmann05]. Garments are composed by a set of Cloth Patterns that has also geometries associated. A Cloth pattern is lying on a Human Segment which has landmarks to identify the right position. The segment of a model is also associated to the Virtual Human Structure description, which will be explained in the next section.

### 3.3.2 Classes for Virtual Human Animation

Virtual human animation could be represented in two parts. The essential features to animate a VH (Skeletal structure), and high level features to provide the VH with a realistic behavior (Animation Control).

### Skeletal Structure

For creating a VH that can be animated it is necessary that it fulfills some requirements as having a minimum articulated structure. This human skeleton definition is based on the H-Anim specification. H-Anim is an efficient representation for skeleton-based animation, adopted by the MPEG-4 standard for Humanoid Virtual Characters animation.

*"The human body consists of a number of segments (such as the forearm, hand and foot) which are connected to each other by joints (such as the elbow, wrist and ankle). In order for an application to animate a humanoid, it needs to obtain access to the joints and alter the joint angles. The application may also need to retrieve information about such things as joint limits and segment masses.*

*An H-Anim file contains a set of Joint nodes that are arranged to form a hierarchy. Each Joint node can contain other Joint nodes, and may also contain a Segment node which describes the body part associated with that joint".*[3]

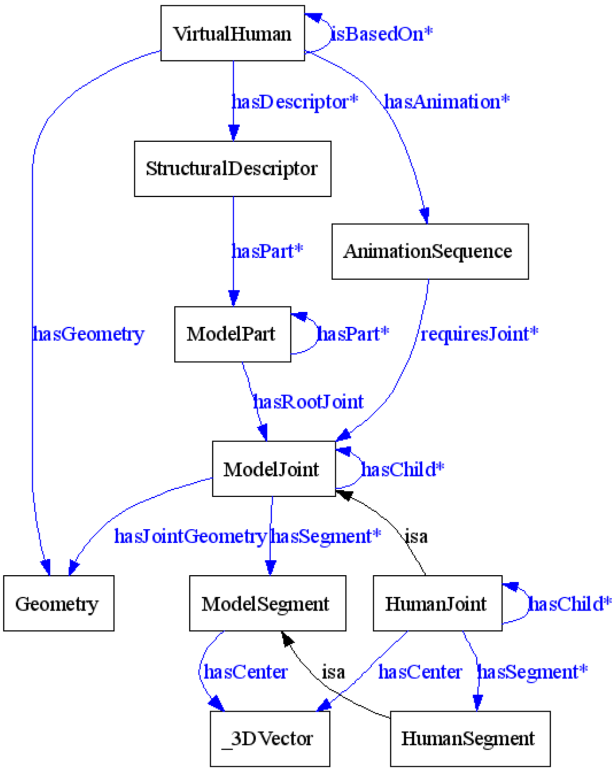


**Figure 8 Virtual Human Skeletal Structure**

This structure allows us to define the complete parts of the VH body. A VH with a complete specification of this structure can be considered as complete. The most interesting question is:

> *Is the VH complete (does it have a skeleton, a hierarchy of body parts, a set of landmarks attached to it)?* which can be formulated as:
> (retrieve (hasDescriptor *this-model* ?descriptor)
> (?descriptor *StructuralDescriptor*)
> (hasPart *this-model* ?part)
> (hasRootJoint ?part ?joint)

[3] http://h-anim.org/Specifications/H-Anim1.1

(?joint *HumanJoint*)
(hasChild ? joint ?jointChild)
(hasGeometry *this-model* ?geometry)
(hasLandmarks ? geometry ?landmark)

In this query the property *hasChild* in class ModelJoint is transitive allowing retrieving recursively all human joints of a VH object. In the presented query we can explore the VH structure but currently we cannot fully answer the question posed. What is missing is the definition of the minimal structure a VH should have to be considered as complete. This is important to know because if the VH does not have all this specifications it cannot be animated.

**Animation Control**

Virtual Human animation could be created by means of pre-recorded animation sequences or behavioral controllers. Animation sequences can be applied to one or many VHs. They could be constrained to some requirements to indicate whether this sequence can be applied to other VHs. A set of rules can be established, e.g. Animation sequences of type MPEG-4 BAP can be applied to any VH with an H-Anim skeleton.

Behavioral animations are controllers that implement an algorithm used to produce a particular behavior on the VH. Behavioral controllers specify the inputs required for the algorithm to work and the outputs (usually animation sequences or specific joint values) it is capable to produce. The class descriptions of the animation sequence and behavior controller are presented in the following figure.

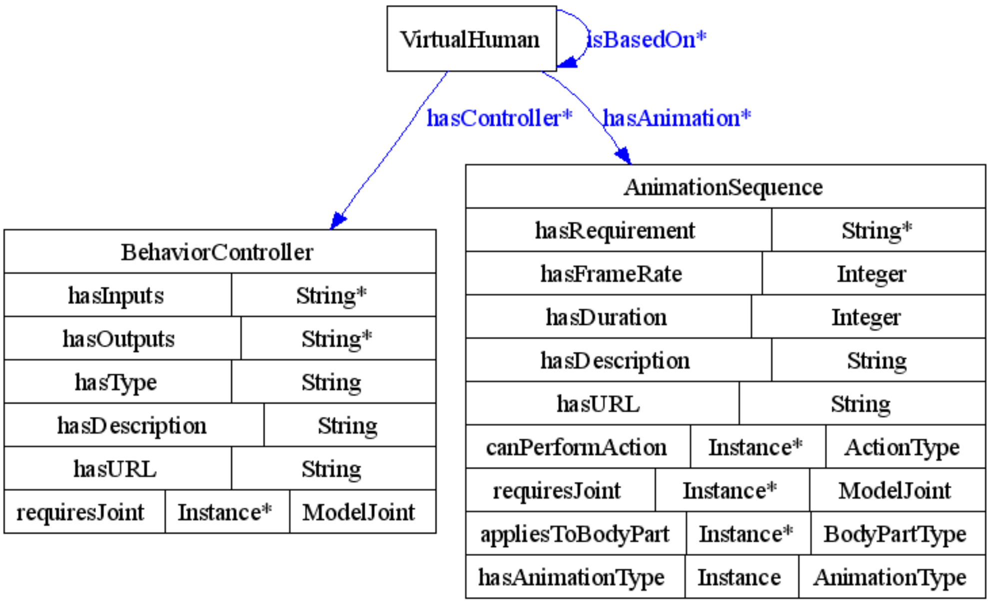


**Figure 9 Animation Control classes**

These behavioral controllers can be of many types and many purposes. A sensorial behavior is a kind of behavioral controller and can be implemented also with various kinds of algorithms with the intention that the VH become capable of perceiving its surroundings (sense of view, audition, and touch).

The Behavioral controllers may use psychological models for a realistic animation. These models are personality and emotions. They are considered as part of the animated virtual human to denote its individuality. As mentioned in the section 2.2.2 there are several models depending on the usage.

In the following figure we include the existence of these models which can be Input for the Behavior controller.

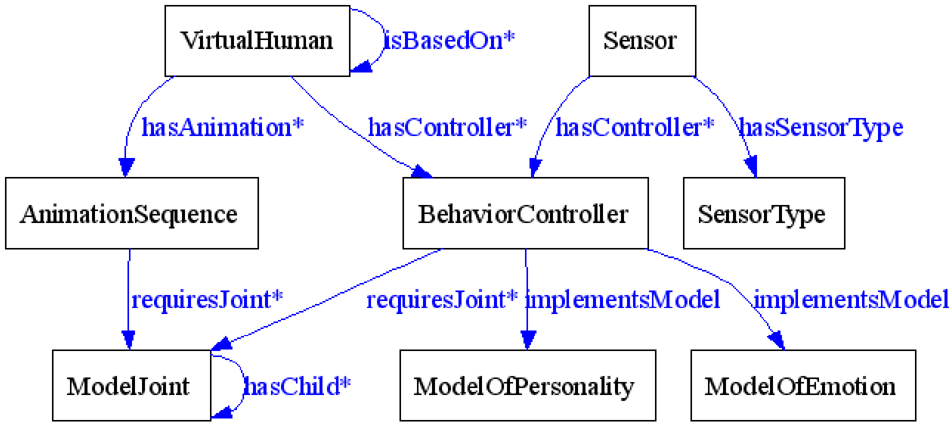


**Figure 10 Behavior Control classes**

Taking into account the mentioned classes we are able to form several competency questions:

*What are the joints affected by this animation sequence?*
(retrieve (requiresJoint *this-animation-sequence* ?joint))

*Are there any animation sequences lasting more than 1 minute?*
(retrieve (hasDuration ?animation <1) (?animation *AnimationSequence*))

*What are the input and output channels of a particular Behavior controller (animation algorithm)?*
(retrieve (hasOutputs *this-controller* ?output) (requiresJoint *this-controller* ?joint))

For making inferences of the VH features, a set of subclasses can be defined; for example, a class describing a VH that can be animated. This is not meant to be used for creating instances directly. Instead it is used for inferring which VH instances are also instances of this class (by checking the Necessary and Sufficient restrictions in the class description).

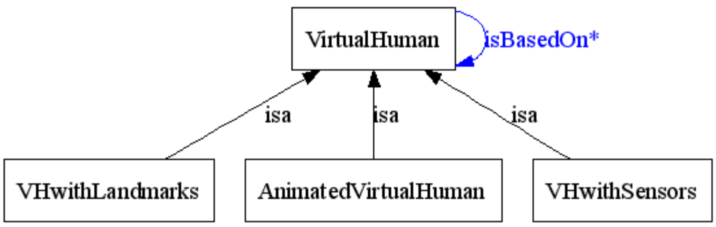


**Figure 11 Inferred Classes of the VH**

Figure 11 presented above shows subclasses of class VirtualHuman that are used to represent specific types of VHs. The following restrictions have been defined:

*VHwithLandmarks* isa VirtualHuman ∃ hasGeometry (Geometry ∩ (∃ hasLandmark Landmark))

*The restrictions for this class describe a Virtual Human which has some Geometry that has some Landmarks.*

*AnimatedVirtualHuman* isa VirtualHuman (∃ hasAnimation AnimationSequence) ∩ (∃ hasDescriptor HAnimSkeleton)
*The restrictions for this class describe a Virtual Human which has at least one Animation Sequence that has at least one Descriptor of type HAnimSkeleton.*

*VHwithSensors* isa VirtualHuman ∃ hasDescriptor (Descriptor ∩ (∃ hasPart (ModelPart ∩ (∃ hasRootJoint (HumanJoint ∩ (∃ hasSegment (HumanSegment ∩ (∃ hasSensor Sensor)))))))
*The restrictions for this class describe a Virtual Human which has at least one Descriptor that has some Model Parts with root joints of type HumanJoint that have some Sensors attached to their Human Segments.*

### 3.3.3 Classes for Virtual Human interactions

As we mentioned in the section 2.3, virtual humans are capable of interacting with the environment, in this case with virtual objects. For handling this, we used the Smart Object approach, which suggests that an object has information about how the user can interact with it.

Giving clues to aid the interaction, the Smart Object class encloses all those objects that can be manipulated by VHs. This class is constituted by a hierarchical collection of nodes. The hierarchical organization specifies the relations between different Geometries and Attributes.

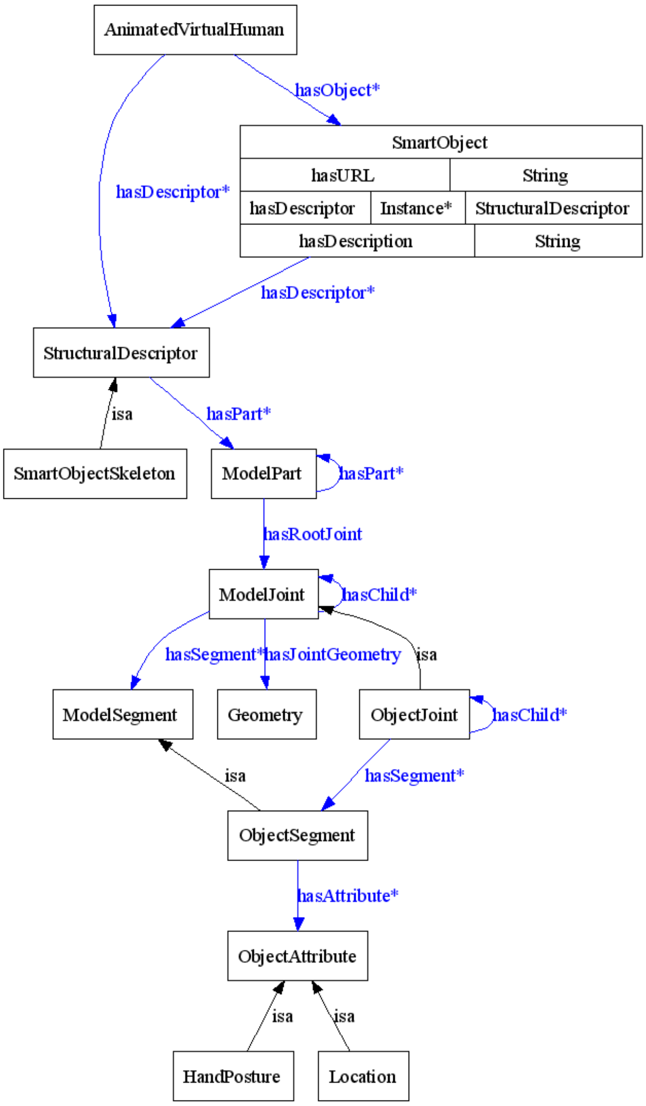


**Figure 12 Smart Object classes**

The attributes of the object that this first version of the ontology has considered is that the object has a structure, a geometry, a location and for interacting with a VH it has a hand posture (for grasping it, for example). The advantage of smart objects is having the geometry of any desired object we have the specification of usage in it. Some related CQs are:

> *What are the characteristics of an object (structure, physical properties, etc.)?*
> (retrieve (hasAttribute *this-smart-object* ?attribute))
> *How can the object be grasped?*
> (retrieve (hasAttribute *this-smart-object* ?attribute) ∩ (?attribute *HandPosture*))

## 3.4 Usage scenarios

### Virtual Characters data repository

A search engine for retrieving VHs and Smart Objects with particular features/functionalities related to animation. As animating characters is a tedious and resource consuming process, animation reusability is highly desirable. This could be implemented inside the repository where users can search for animations with specific conditions like a mature woman running, and using the morphological descriptors and attributes of the animation. The result will be different types of files corresponding just to the animation, or to the animation with the model.

**Modeling a data repository**

A place where a modeler and animator could find VH shapes in order to use them to model new VHs, improve or reconstruct existing ones. The ontology could help to manage an online shop to build custom a 3D mannequin and select and fit clothes and accessories for visualizing on the mannequin. This means that it will be very useful to know the suitable clothes and accessories to a specific VH by matching the corresponding landmarks.

**Shape recognition/extraction/analysis**

A knowledge base that is able to answer competency questions linked to low level features of the VH shape (landmarks, topological graphs, and so on). Target users would include researchers working on algorithms for recognizing features on a shape representing a virtual/real human. Data would be used on ergonomics studies, computer vision algorithms, etc.

Let us consider how an instance of a Virtual Human could be represented with the proposed ontology. We consider a soccer player for illustration. To model such an instance we need a 3D human body shape with a full articulated skeleton attached. The required associated clothes would include at least a short, a tee-shirt and sockets connected to the appropriate parts of the body. The additional accessories are the soccer shoes. The basic associated smart object is a ball including all the possible interactions between the player and the ball and the related landmarks for manipulation and interactions. The associated animation sequences should cover all the possible actions that a soccer player can perform: running, jumping, taking the ball, drop the ball, etc.

It would then be easy to derive instances of the soccer player to create a soccer team by modifying the morphology descriptors to generate different players, retargeting the animation sequences, fitting the clothes and accessories. It would also be possible to derive specialized instances such as a goal keeper by attaching additional accessories such as gloves and including additional animation sequences (as a goal keeper as some specific actions and interactions with the ball).

# 4 REFERENCES

[BookStein97] F. L. BookStein. Morphometric Tools for Landmark Data: Geometry and Biology. Cambridge Press, 1997.

[Dey05] T. K. Dey. Chapter in Geometric and Algorithmic Aspects of Computer-Aided Design and Manufacturing, eds Janardan, Smid, Dutta. DIMACS series in Discrete Mathematics and Theoretical Computer Science, Volume 67, 2005.

[Daanen98] H. A. M. Daanen and G. J. van de Water. Whole body scanners. Displays, 19(3):111–120, November 1998.

[VHOnto] M. Gutierrez, D. Thalmann, F. Vexo, L. Moccozet, N. Magnenat-Thalmann, M. Mortara, M. Spagnuolo, An Ontology of Virtual Humans: incorporating semantics into human shapes, Workshop towards Semantic Virtual Environments (SVE05), March 2005, Villars, Switzerland, pages 57-67.

[Fuhrmann05] A. Fuhrmann, C. Groß, A. Weber *Ontologies for Virtual Garments.* Workshop towards Semantic Virtual Environments (SVE 2005), pp. 101-109, Villars, Switzerland, March 2005

[Gruber93] Gruber T. R.,1993. *A Translation Approach to Portable Ontology Specifications.* In Knowledge Acquisition. vol. 6, no. 2, pp. 199-221

[Gtz_IJCAT05] M. Gutiérrez, F. Vexo, D. Thalmann, *Semantics-based representation of Virtual Environments*, International Journal of Computer Applications in Technology, 2005 Vol.23, No 2/3/4

[Gtz_SVE05] M. Gutiérrez, D. Thalmann, and F. Vexo. *Semantic virtual environments with adaptive multimodal interfaces*. In Proceedings of the 11th International Conference on Multimedia Modelling (MMM2005), pages 277–283, 2005.

[Gruninger et al.] M. Gruninger and M. Fox. *Methodology for the Design and Evaluation of Ontologies*. In: Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal.

[h-anim] H-Anim Working Group. ISO/IEC FCD 19774:200x, Humanoid Animation, Annex B, Feature points for the human body. http://www.h-anim.org/.

[Mortara03] M. Mortara, G. Patane, M. Spagnuolo, B. Falcidieno, and J. Rossignac. Blowing bubbles for multi-scale analysis and decomposition of triangle meshes. Algorithmica, 38(1):227–248, 2003.

[Kallmann99] M. Kallmann, D. Thalmann, Direct 3D *Interaction with Smart Object*, Proc. ACM VRST 99, London

[Protege05] *Protégé*. (c) 2005 Stanford medical informatics. http://protege.stanford.edu/index.html .