

Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Συνθετική κίνηση εικονικών πρακτόρων για το ευφυές εικονικό περιβάλλον REVE Worlds
Όνοματεπώνυμο Φοιτητή	Γκόλφω-Ελένη Βασιλείου
Πατρώνυμο	Αντώνιος
Αριθμός Μητρώου	ΜΠΣΠ09013
Επιβλέπων	Θεμιστοκλής Παναγιωτόπουλος, Καθηγητής

Ημερομηνία Παράδοσης

Δεκέμβριος 2011

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΑΙΑ

Τριμελής Εξεταστική Επιτροπή

Παναγιωτόπουλος Θεμιστοκλής
Καθηγητής

Τσιχριντζής Γεώργιος
Καθηγητής

Φούντας Ευάγγελος
Καθηγητής

Περίληψη

Ο σχεδιασμός και η δημιουργία ενός εικονικού πράκτορα είναι ένα πολυσύνθετο πρόβλημα το οποίο ερευνάται εδώ και χρόνια. Όταν ένας πράκτορας σε ένα εικονικό περιβάλλον έχει ανθρώπινη μορφή, μεγάλο ρόλο στη ρεαλιστική αναπαράσταση της συμπεριφοράς του παίζουν, εκτός από τη νοημοσύνη του η εμφάνιση του και ο τρόπος που κινείται και αντιδράει στα ερεθίσματα που δέχεται από το περιβάλλον του.

Αυτή η διπλωματική παρέχει μια επέκταση στο ευφυές εικονικό περιβάλλον REVE Worlds. Οι εικονικοί πράκτορες που διαθέτει το REVE Worlds έχουν ως σώματα απλά τρισδιάστατα μοντέλα με περιορισμένες δυνατότητες κίνησης. Σκοπός αυτής της διατριβής είναι σχεδίαση και υλοποίηση μιας συστηματικής υποδομής που θα επιτρέπει την εύκολη δημιουργία και ενσωμάτωση κινήσεων και μοντέλων για τους πράκτορες.

Αφού έγινε διαχωρισμός της έννοιας των εικονικών πρακτόρων σε τρία επίπεδα (μοντέλο, κίνηση, συμπεριφορά) και παρουσιάστηκαν προηγούμενες σχετικές προσεγγίσεις και βιβλιογραφία, επιλέχθηκε το πρότυπο ανθρωπόμορφης μοντελοποίησης H-ANIM ως η κατάλληλη βάση για το REVE Worlds.

Για την επίδειξη της ορθής λειτουργίας του πλαισίου μελετήθηκαν δύο περιπτώσεις χρήσης. Ένα animation για περπάτημα και ένα animation κίνησης χεριού για το πιάσιμο κάποιου αντικειμένου. Ο έλεγχος των κινήσεων γίνεται σε πραγματικό χρόνο μέσω εντολών του interface του REVE Worlds.

Το αποτέλεσμα ήταν η σχεδίαση και η υλοποίηση μιας υποδομής που επιτρέπει στο REVE Worlds να εκμεταλλεύεται την ευελιξία που προσφέρει το H-ANIM.

Abstract

The design and creation of a virtual agent is a complicated problem that has been the subject of research for many years. When an agent in a virtual environment has a human-like form, it's not only appearance and intelligence that play a big part in it's realistic representation, but also the way it moves and reacts to the stimuli it receives from it's environment.

This thesis provides an extension to the virtual agents of the REVE Worlds intelligent virtual environment. The virtual agents available in REVE Worlds currently have simple 3d models with limited movement abilities as bodies. The goal of this thesis is to design and implement a framework that would allow the easy creation and integration of animated bodies for the agents.

After the concept of the virtual agent was divided in three layers (model, animation, behavior) and existing approaches and literature were presented, the H-ANIM standard for humanoid modeling was selected as a suitable base for the framework.

The proper functioning of the framework is demonstrated through two use cases. An animation for walking and an animation for moving the right arm to touch an object. These are controlled in real-time within the REVE Worlds interface by using appropriate commands.

The result was the design and implementation of a framework that enables REVE Worlds to take advantage of the flexibility offered by H-ANIM.

Γκόλφω-Ελένη Βασιλείου
Προηγμένα Συστήματα Πληροφορικής

Copyright © Γκόλφω-Ελένη Βασιλείου, 2011.
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΔΙΑ

Στον Πάνο

ΕΥΧΑΡΙΣΤΙΕΣ

Ευχαριστώ την οικογένειά μου για την ηθική υποστήριξη της.

Ευχαριστώ τον επιβλέποντα της πτυχιακής εργασίας κ.Παναγιωτόπουλο καθώς και τον κ. Αναστασάκη για την υποστήριξη,την καθοδήγηση και τις συμβουλές τους για να φτάσει αυτή η εργασία στο τέλος της. Τέλος ευχαριστώ τον Νίκο Αβραδίνη τις πληροφορίες του σχετικά με την μηχανή UNREAL.

Τέλος, θέλω να ευχαριστήσω φίλους και γνωστούς για το κουράγιο που μου έδωσαν ώστε να συνεχίζω κάθε φορά.

Γκόλφω-Ελένη Βασιλείου,

Πειραιάς, 2011

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1	ΕΙΣΑΓΩΓΗ.....	10
1.1	ΕΙΣΑΓΩΓΗ	10
1.2	Η ΔΙΑΡΘΡΩΣΗ ΤΗΣ ΔΙΑΤΡΙΒΗΣ	10
2	ΒΑΣΙΚΕΣ ΈΝΝΟΙΕΣ/ΑΝΑΣΚΟΠΗΣΗ.....	12
2.1	ΕΙΣΑΓΩΓΗ	12
2.2	ΕΠΙΠΕΔΟ ΜΟΝΤΕΛΟΥ	13
2.2.1	<i>Εισαγωγή.....</i>	<i>13</i>
2.2.2	<i>H-ANIM [5].....</i>	<i>14</i>
2.2.3	<i>MPEG-4 [6].....</i>	<i>14</i>
2.2.4	<i>MPEG-4 FBA [7].....</i>	<i>15</i>
2.3	ΕΠΙΠΕΔΟ ΚΙΝΗΣΗΣ.....	16
2.3.1	<i>Κινηματικές μέθοδοι.....</i>	<i>16</i>
2.3.2	<i>Δυναμικές μέθοδοι.....</i>	<i>16</i>
2.3.3	<i>Data Driven Animation.....</i>	<i>16</i>
2.3.4	<i>Procedural Animation.....</i>	<i>17</i>
2.4	ΕΠΙΠΕΔΟ ΣΥΜΠΕΡΙΦΟΡΑΣ.....	17
2.4.1	<i>AML: Avatar Markup Language [14].....</i>	<i>17</i>
2.4.2	<i>CML: Character Markup Language [15].....</i>	<i>18</i>
2.4.3	<i>PAR: Parametrized Action Representation [16].....</i>	<i>18</i>
2.4.4	<i>STEP: Scripted Technology for Embodied Persona [17].....</i>	<i>18</i>
2.4.5	<i>BML: Behavior Markup Language [19][20].....</i>	<i>18</i>
2.4.6	<i>VHML: Virtual Human Markup Language [21].....</i>	<i>18</i>
2.4.7	<i>Σύγκριση χαρακτηριστικών των γλωσσών συμπεριφοράς [22].....</i>	<i>19</i>
2.5	REAL-TIME CHARACTER ENGINES [23].....	20
3	ΑΝΑΛΥΣΗ.....	23
3.1	ΤΟ ΠΡΟΤΥΠΟ H-ANIM.....	23
3.2	ΠΛΕΟΝΕΚΤΗΜΑΤΑ H-ANIM.....	23
3.3	ΠΑΡΑΔΕΙΓΜΑΤΑ ΧΡΗΣΕΩΝ ΤΟΥ H-ANIM.....	24
3.4	ΓΕΩΜΕΤΡΙΑ H-ANIM ΣΩΜΑΤΟΣ [5].....	24
3.4.1	<i>Κόμβοι του H-ANIM.....</i>	<i>24</i>
3.4.2	<i>Levels of Articulation [29].....</i>	<i>25</i>
4	ΣΧΕΔΙΑΣΜΟΣ.....	27
4.1	ΕΙΣΑΓΩΓΗ	27
4.2	Η ΠΛΑΤΦΟΡΜΑ REVE ΚΑΙ ΤΟ REVEWORLDS [33].....	27
4.2.1	<i>Εικονικοί πράκτορες στο REVEWorlds[35].....</i>	<i>28</i>

4.3	ΔΟΜΗ	30
5	ΥΛΟΠΟΙΗΣΗ.....	31
5.1	ΕΙΣΑΓΩΓΗ	31
5.1.1	Σύντομη περιγραφή του X3D[30][31].....	31
5.2	ΑΡΧΕΙΑ X3D	32
5.2.1	Περιγραφή αρχείου X3d για το σώμα	32
5.2.2	Περιγραφή αρχείου X3d για το animation	33
5.2.3	Παράδειγμα Route	34
5.2.4	Παράδειγμα Animator.....	34
5.2.5	Script για την κίνηση του χεριού	35
5.3	ΑΡΧΕΙΑ JAVA	35
5.3.1	Περιγραφή της κλάσης HanimBody[35].....	35
5.3.2	Περιγραφή της κλάσης για τους animators[35].....	38
5.3.3	Παράδειγμα ενέργειας.....	39
6	CASE STUDY	41
6.1	ΠΑΡΑΔΕΙΓΜΑΤΑ ΛΕΙΤΟΥΡΓΙΑΣ	41
6.1.1	Walk Animation.....	42
6.1.2	Touch Animation.....	43
6.2	Η-ANIM ΠΡΑΚΤΟΡΕΣ ΣΤΟ REVEWORLDS	44
6.2.1	Walk Animation.....	44
6.2.2	Σύγκριση του Walk Animation για διαφορετικές ταχύτητες.....	45
7	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	48
7.1	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	48
7.2	ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ	48
8	ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ.....	49

ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ

Εικόνα 1:Τα επίπεδα από τα οποία αποτελείται ένας εικονικός χαρακτήρας	13
Εικόνα 2:Δείγμα σκηνης πολυμέσων βασισμένης στο πρότυπο MPEG-4 [41]	14
Εικόνα 3: Τα μέρη από τα οποία αποτελείται ένα εικονικό σώμα κατά το MPEG-4 [42].....	15
Εικόνα 4:H-ANIMATOR [42]	23
Εικόνα 5: Όλες οι αρθρώσεις που προσφέρει το H-ANIM[42]	25
Εικόνα 6: Οι αρθρώσεις του LOA-1 και οι θέσεις τους σε ένα ανθρωπόμορφο μοντέλο[41]	26
Εικόνα 7:Μοντέλο δράσης εικονικού πράκτορα[42]	29
Εικόνα 8:Το REVEWorlds και ένας εικονικός πράκτορας.[42]	29
Εικόνα 9:Η δομή για την ενσωμάτωση κίνησης	30
Εικόνα 10:Η δομή του X3d αρχείου για το μοντέλο.....	32
Εικόνα 11: Η δομή του αρχείου X3d για το animation	33
Εικόνα 12:Ο Boxman	41
Εικόνα 13: Ο χρήστης μπορεί να επιλέξει animation (περπάτημα ή κίνηση του χεριού) ..	42
Εικόνα 14:Walk Animation	42
Εικόνα 15:Κίνηση του χεριού σε διάφορες θέσεις.....	43
Εικόνα 16: Το περιβάλλον REVE Worlds με τη δυνατότητα φόρτωσης ενός σώματος H- ANIM.	44
Εικόνα 17: Οι εντολές για περπάτημα στο CLI	45
Εικόνα 18: Το Walk Animation στο REVE Worlds	45
Εικόνα 19:Οι τρεις πράκτορες στην εκκίνησή τους στα 0 sec	46
Εικόνα 20:Οι τρεις πράκτορες στη χρονική στιγμή 1 sec	46
Εικόνα 21:Οι τρεις πράκτορες στη χρονική στιγμή 5 sec	46
Εικόνα 22:Οι τρεις πράκτορες στη χρονική στιγμή 8 sec	47
Εικόνα 23:Οι τρεις πράκτορες στη χρονική στιγμή 13 sec	47
Εικόνα 24:Στα 18 sec έχει τερματίσει και ο τελευταίος πράκτορας.....	47

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Εισαγωγή

Αυτή η διατριβή αντιμετωπίζει το πρόβλημα της συνθετικής κίνησης εικονικών ανθρώπων σε εικονικά περιβάλλοντα και έχει ως σκοπό την ανάπτυξη ενός πλαισίου για την κίνηση των εικονικών πρακτόρων του εικονικού περιβάλλοντος REVE. Οι κινήσεις που θα εκτελούν οι πράκτορες θα είναι παραλλαγές κάποιων πρότυπων κινήσεων που θα έχουν δημιουργηθεί από πριν και των οποίων οι παράμετροι θα ρυθμίζονται σε πραγματικό χρόνο. Έτσι, οι τελικές κινήσεις του πράκτορα θα είναι προσαρμοσμένες στις περιστάσεις που προκύπτουν κάθε φορά στο περιβάλλον του και στις απαιτήσεις του χρήστη, προσδίδοντας στην αληθοφάνεια της αναπαράστασης.

Για να επιτευχθεί αυτό πρέπει να χρησιμοποιηθούν τρισδιάστατα μοντέλα που προσομοιάζουν το ανθρώπινο σώμα και να δημιουργηθούν κινήσεις προσαρμοσμένες στη σκελετική τους δομή. Τα αρχεία των μοντέλων και των κινήσεων θα διαβάζονται από το REVE Worlds και θα συνθέτουν την απεικόνιση του πράκτορα, με τον έλεγχό του να γίνεται επίσης μέσω του εικονικού περιβάλλοντος.

1.2 Η διάρθρωση της διατριβής

Η διατριβή ακολουθεί την παρακάτω δομή:

- Στο δεύτερο κεφάλαιο, γίνεται βιβλιογραφική ανάφορα στην υπάρχουσα έρευνα του χώρου που κινείται η διατριβή. Παρουσιάζονται και αναλύονται βασικές έννοιες και πρότυπα που αφορούν την εμφάνιση, την κίνηση και τη συμπεριφορά των εικονικών πρακτόρων.
- Μετά ακολουθεί η ανάλυση του προτύπου που χρησιμοποιήθηκε για την υλοποίηση της τρισδιάστατης κίνησης και αναφέρονται οι λόγοι που επιλέχθηκε το H-ANIM ως πρότυπο υλοποίησης.

- Το κεφάλαιο που αναφέρεται στο σχεδιασμό περιέχει μία σύντομη παρουσίαση του REVE Worlds και σχεδιαγράμματα της υποδομής με βάση την οποία Η-ANIM σώματα μπορούν να αξιοποιηθούν από την πλατφόρμα REVE.
- Στο πέμπτο κεφάλαιο της διατριβής παρουσιάζεται η υλοποίηση. Επεξηγούνται τα σημαντικότερα κομμάτια του κώδικα και δίνονται μερικά παραδείγματα λειτουργίας.
- Το έκτο κεφάλαιο περιέχει παραδείγματα της χρήσης της υποδομής στο REVE Worlds.
- Η διατριβή κλείνει με συμπεράσματα και τρόπους με τους οποίους μπορεί να αξιοποιηθεί και να επεκταθεί περαιτέρω η υποδομή που παρουσιάστηκε.

ΚΕΦΑΛΑΙΟ 2

Βασικές Έννοιες/Ανασκόπηση

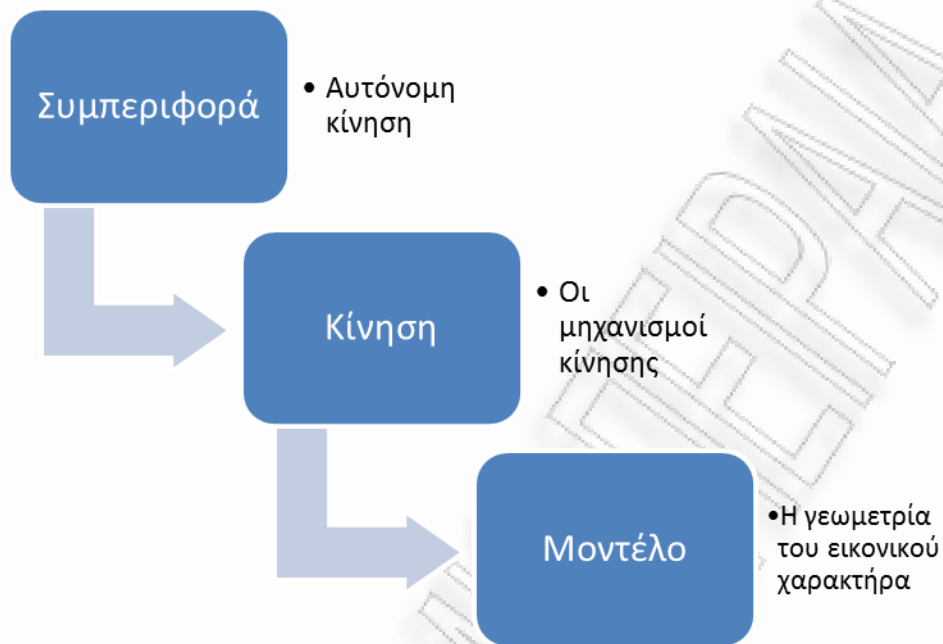
2.1 Εισαγωγή

Οι εικονικοί πράκτορες πρέπει να εκτελούν συγκεκριμένες εργασίες και να είναι όσο το δυνατόν περισσότερο αληθοφανείς και αλληλεπιδραστικοί. Τα τμήματα που πρέπει να υλοποιηθούν για να κατασκευαστεί ένας ολοκληρωμένος εικονικός ανθρωπόμορφος πράκτορας σύμφωνα με τον Thalmann [1] είναι τα εξής:

- Αναπαράσταση προσώπου και σώματος
- Λειτουργίες του Avatar
- Έλεγχος κίνησης
- Συμπεριφορά υψηλού επιπέδου
- Αλληλεπίδραση με αντικείμενα
- Ενδοεπικοινωνία
- Αλληλεπίδραση με το χρήστη
- Συνεργατικά Εικονικά Περιβάλλοντα
- Πλήθη
- Απόδοση (rendering)

Οι εικονικοί πράκτορες λειτουργούν σε πραγματικό χρόνο και κατά συνέπεια παρουσιάζονται δυσκολίες και περιορισμοί σε σχέση με τα μοντέλα τα οποία έχουν συγκεκριμένο animation που έχει σχεδιαστεί από πριν. Σε αντίθεση με τα προκατασκευασμένα μοντέλα, ο αριθμός των πολυγώνων του πράκτορα και οι μέθοδοι που χρησιμοποιούνται για το animation του σώματος του περιορίζονται από το hardware του κάθε υπολογιστή που τα χρησιμοποιεί [2]. Έτσι, δεν μπορούσαν να χρησιμοποιηθούν πολύπλοκες τεχνικές, που θα έδιναν πιο ρεαλιστικά αποτελέσματα, αλλά αυτό έχει αλλάξει με την πρόοδο της τεχνολογίας. Άλλη πρόκληση που παρουσιάζει το animation των εικονικών πρακτόρων, είναι ότι η κίνηση τους δε μπορεί να προβλεφθεί με ακρίβεια και εξαρτάται από το περιβάλλον τους και τις ενέργειες του χρήστη.

Ένας εικονικός ανθρωπόμορφος χαρακτήρας μπορεί να θεωρηθεί ότι χωρίζεται σε τρία επίπεδα.



Εικόνα 1: Τα επίπεδα από τα οποία αποτελείται ένας εικονικός χαρακτήρας

Τα ανθρώπινα μοντέλα συνήθως αναπαρίστανται από ένα αριθμό τμημάτων τα οποία συνδέονται μεταξύ τους με αρθρώσεις. Η κίνηση τους επιτυγχάνεται με την αλλαγή της στάσεως του σώματος με διάφορους τρόπους. Όταν ένας χαρακτήρας είναι αυτόνομος, όπως ένας εικονικός πράκτορας, και εκτελεί πιο πολύπλοκες κινήσεις τότε εκδηλώνει συμπεριφορά.

2.2 Επίπεδο Μοντέλου

2.2.1 Εισαγωγή

Μπορεί να χρησιμοποιηθεί οποιοδήποτε τρισδιάστατο μοντέλο. Τα μοντέλα συνήθως σχεδιάζονται στο χέρι και για τη δημιουργία ρεαλιστικών μοντέλων μπορούν να χρησιμοποιηθούν ανθρωπομετρικά δεδομένα, όπως αυτά της βάσης δεδομένων CAESAR [3]. Για την αυτόματη δημιουργία πολλών, διαφορετικών εικονικών ανθρώπων έχει προταθεί η δημιουργία τους με εξελικτικούς αλγορίθμους [4]

Η ανάγκη για επαναχρησιμοποίηση των μοντέλων, ανεξάρτητα από το εργαλείο σχεδιασμού, οδήγησε στην ανάγκη κάποιων προτύπων. Τα πιο διαδεδομένα πρότυπα είναι το H-ANIM του Web 3d Consortium και το MPEG-4

2.2.2 H-ANIM [5]

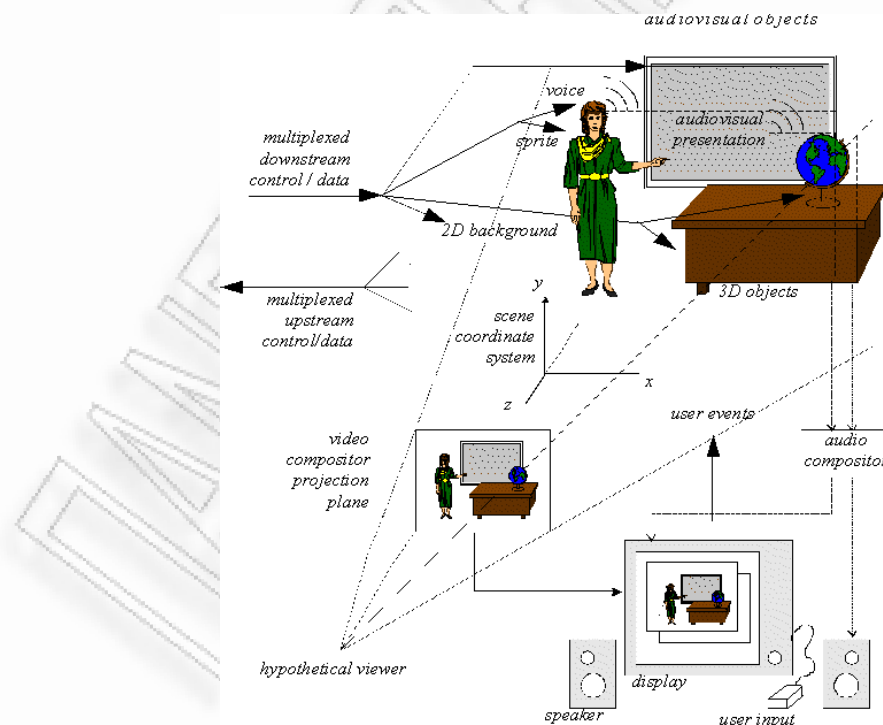
Το H-ANIM σχηματίστηκε από το Web3D Consortium για να παρέχει στους προγραμματιστές κάποιους κανόνες για την ονομασία των μερών του ανθρώπινου σώματος. Τα animations για μοντέλα που βασίζονται στο H-ANIM μπορούν να δημιουργηθούν με διάφορες τεχνικές (keyframing, inverse kinematics, performance animation systems κτλ).

Το H-ANIM έχει τρεις κύριους στόχους. Να επιτύχει :

- *Συμβατότητα*: Τα μοντέλα να εμφανίζονται σε οποιονδήποτε browser υποστηρίζει VRML ή X3d.
- *Ευελιξία*: Δεν υπάρχει κάποιος περιορισμός για το είδος των εφαρμογών που θα χρησιμοποιήσουν τα μοντέλα.
- *Απλότητα*: Οι προδιαγραφές του προτύπου περιλαμβάνουν μόνο ότι είναι απολύτως απαραίτητο.

2.2.3 MPEG-4 [6]

Το πρότυπο MPEG-4, σε αντίθεση με τα προηγούμενα πρότυπα MPEG, εισάγει την έννοια των οπτικών και ακουστικών αντικειμένων που απαρτίζουν μια σκηνή πολυμέσων. Το MPEG-4 στοχεύει στο να επιτύχει συμπίεση του κάθε τύπου αντικειμένου, υβριδική κωδικοποίηση των συνθετικών και φυσικών αντικειμένων, προσβασιμότητα μέσω διαφόρων δικτύων και αλληλεπιδραστικότητα.



Εικόνα 2: Δείγμα σκηνής πολυμέσων βασισμένης στο πρότυπο MPEG-4 [41]

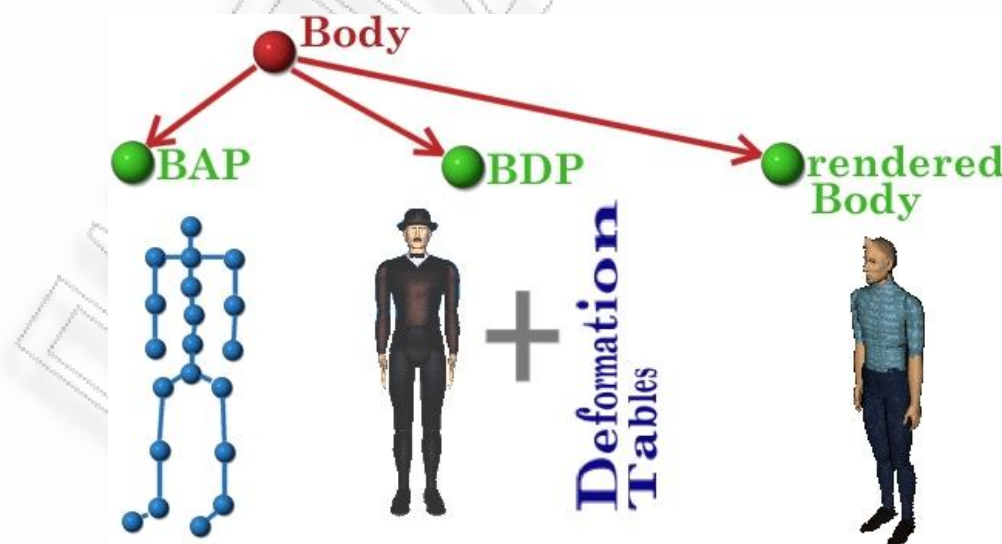
Για να προσδιορίσει τη χωροχρονική τοποθεσία ενός αντικειμένου στη σκηνή, το πρότυπο ορίζει μια γλώσσα που ονομάζεται BIFS (Binary Format for Scenes). Η BIFS δανείζεται στοιχεία από τη VRML, όπως την περιγραφή της σκηνής σαν ένα ιεραρχημένο γράφημα, τα animations (που βασίζονται σε παρεμβολείς (Interpolators)), τα events που στέλνονται στους κόμβους μέσω Routes και τη χρήση των sensors για αλληλεπίδραση με το περιβάλλον. Πέρα από αυτά, εισάγει κάποια νέα στοιχεία όπως μηχανισμούς συμπίεσης, streamed animation, ενσωμάτωση 2d αντικειμένων και εξελιγμένο χρονικό έλεγχο.

2.2.4 MPEG-4 FBA [7]

Το MPEG-4 ορίζει το FBA (Facial and Body Animation) για να περιγράψει τη γεωμετρία και το animation ενός εικονικού χαρακτήρα.

Υπάρχουν δύο σύνολα παραμέτρων. Το πρώτο αποτελείται από τα FDP (Face Definition Parameters) και BDP (Body Definition Parameters) τα οποία περιγράφουν τη γεωμετρία του προσώπου και του σώματος αντίστοιχα. Το BDP βασίζεται στο H-ANIM. Αυτοί οι παράμετροι επιτρέπουν στον αποκωδικοποιητή να δημιουργήσει ένα μοντέλο FBA.

Το δεύτερο σύνολο παραμέτρων ορίζει το animation του προσώπου και του σώματος, δηλαδή τα FAP (Face Animation Parameters) και BAP (Body Animation Parameters). Τα BAP χρησιμοποιούνται για τη σύνθεση κινήσεων. Για να παραχθούν κινήσεις αλλάζονται οι γωνίες των αρθρώσεων του σώματος με βάση τα δεδομένα που περιέχονται σε ένα BAP stream. Ένα BAP stream μπορεί να περιέχει μέχρι 296 παραμέτρους που περιγράφουν την τοπολογία ενός σκελετού.



Εικόνα 3: Τα μέρη από τα οποία αποτελείται ένα εικονικό σώμα κατά το MPEG-4 [42]

2.3 Επίπεδο κίνησης

Οι μέθοδοι για αναπαράσταση κίνησης μπορούν να χωριστούν σε δύο κατηγορίες, κινηματικές και δυναμικές [8]. Οι δύο τρόποι υλοποίησης κίνησης που παρουσιάζονται εδώ είναι οι Data Driven Animation (Κίνηση οδηγούμενη από τα δεδομένα) και Procedural Animation (Διαδικαστική κίνηση).

2.3.1 Κινηματικές μέθοδοι

Η κινηματική ασχολείται με τη θέση, την επιτάχυνση και την ταχύτητα των σωμάτων χωρίς να θεωρείται ότι έχουν κάποια μάζα. Με τη χρήση της ευθείας κινηματικής (forward kinematics) μπορούμε να βρούμε την τελική θέση του αντικειμένου δίνοντας ένα σύνολο παραμέτρων, ενώ με τη χρήση της αντίστροφης κινηματικής (inverse kinematics) μπορούμε να βρούμε τις παραμέτρους που χρειάζεται να δώσουμε σε ένα μοντέλο για να βρεθεί σε κάποια συγκεκριμένη θέση.

2.3.2 Δυναμικές μέθοδοι

Βασίζονται στους νόμους του Νεύτωνα για την κίνηση και συσχετίζουν τα αίτια της κίνησης (τις δυνάμεις που ασκούνται) με την επιτάχυνση ενός σώματος λαμβάνοντας υπόψη τη μάζα του. Υπάρχουν κι εδώ οι αντίστοιχες μέθοδοι της ευθείας δυναμικής και της αντίστροφης δυναμικής. Όταν χρησιμοποιούμε μια ευθεία δυναμική μέθοδο βρίσκουμε τη θέση, τον προσανατολισμό, τη θέση και την ταχύτητα που αντιστοιχούν σε κάποιες δυνάμεις και ροπή ενώ με την αντίστροφη δυναμική γίνεται η αντίθετη διαδικασία.

Η υπολογιστική πολυπλοκότητα των παραπάνω μεθόδων, με αύξουσα σειρά, είναι η εξής :

- Ευθεία Κινηματική
- Ευθεία Δυναμική
- Αντίστροφη Κινηματική
- Αντίστροφη Δυναμική

Καμία τεχνική δε μπορεί να δώσει τέλεια αποτελέσματα για όλο το σώμα, και συνήθως χρησιμοποιείται ένας συνδυασμός των παραπάνω. Δημιουργούνται συνήθως με τεχνικές που κατατάσσονται είτε στο data-driven animation, είτε στο procedural animation.[9]

2.3.3 Data Driven Animation

Είναι ο πιο συχνός τρόπος υλοποίησης animation. Τα animations κατασκευάζονται από πριν με motion capture ή σχεδιάζονται στο χέρι με τη χρήση ειδικού λογισμικού. Τα δεδομένα του κάθε συνόλου κίνησης αποτελούνται από μια σειρά keyframes που περιέχουν τιμές για τις παραμέτρους του σώματος. Το πρόβλημα με αυτή τη μέθοδο είναι ότι δεν είναι αρκετά ευέλικτη

γιατί είναι δύσκολο για τις λεπτομέρειες των keyframes να ρυθμιστούν σε πραγματικό χρόνο . Αυτό το πρόβλημα είναι θέμα έρευνας τα τελευταία χρόνια, κι έχουν γίνει προσπάθειες αντιμετώπισής του με τη χρήση βάσεων δεδομένων που περιέχουν πολλές κινήσεις [10] ή με γραφήματα κίνησης που κωδικοποιούν τους τρόπους με τους οποίους τα αποθηκευμένα animations μπορούν να ξαναχρησιμοποιηθούν[11].

2.3.4 Procedural Animation

Για κίνηση που ανταποκρίνεται με μεγαλύτερη ακρίβεια στις απαιτήσεις του εκάστοτε περιβάλλοντος και τη γεωμετρία του χαρακτήρα προτείνεται ως λύση το animation που παράγεται αλγοριθμικά σε πραγματικό χρόνο για να ανταποκρίνεται σε συγκεκριμένες προϋποθέσεις. Με αυτό το τρόπο γίνεται εφαρμογή της Αντίστροφης Κινηματικής (Inverse Kinematics) . Η γωνία περιστροφής των αρθρώσεων καθορίζεται με βάση την τελική θέση που πρέπει να βρεθεί κάποιο άκρο του σώματος (π.χ. χέρι ή πόδι). Χρησιμοποιείται συνήθως για το περπάτημα σε ανώμαλο έδαφος (υπολογισμός της σωστής θέσης της πατούσας σε κάθε βήμα) ή το πιάσιμο αντικειμένων. Εκτός από αυτά όμως, έχουν γίνει και προσπάθειες για τη χρήση της δυναμικής του ανθρώπινου σώματος (ragdoll physics)[12][13] έτσι ώστε η κίνηση του κάθε μέλους του σώματος να είναι συνδεδεμένη με την κίνηση των υπόλοιπων μελών.

2.4 Επίπεδο συμπεριφοράς

Οι εικονικοί πράκτορες πρέπει να έχουν τη δυνατότητα να λειτουργούν αυτόνομα σε πραγματικό χρόνο και να εκδηλώνουν συμπεριφορά που τους επιτρέπει να αλληλεπιδρούν με το περιβάλλον τους, χωρίς να χρειάζεται όλες οι λεπτομέρειες της κίνησής του να έχουν σχεδιαστεί από πριν.

Η συμπεριφορά ενός πράκτορα περιγράφεται συνήθως με μια υψηλού επιπέδου scripting γλώσσα βασισμένη στην XML, η οποία είναι ευρέως διαδεδομένη χάρη στη δομή της που είναι εξίσου ευανάγνωστη από ανθρώπους και μηχανήματα. Μερικά παραδείγματα τέτοιων γλωσσών είναι η BML, AML, CML, MURML, XSTEP, VHML και APML.

2.4.1 AML: Avatar Markup Language [14]

Περιγράφει ένα ενιαίο πλαίσιο για μετατροπή κειμένου σε ομιλία, κίνηση πρόσωπο και κίνηση σώματος με τον κατάλληλο συγχρονισμό μεταξύ τους. Το AML δεν εξαρτάται από κάποιες συγκεκριμένες παραμέτρους, όπως το MPEG-4, για κίνηση σε χαμηλό επίπεδο, χρησιμοποιείται εύκολα από ευφυείς πράκτορες, είναι επεκτάσιμο και υποστηρίζει seamless animations.

2.4.2 CML: Character Markup Language [15]

Υπάρχουν αρκετά λογισμικά και μέθοδοι για τη δημιουργία animation και αρκετές μηχανές δημιουργίας συμπεριφοράς και συναισθημάτων, αλλά δεν υπάρχει κάποιο κοινό συνδεδειγμένο επίπεδο ανάμεσα τους. Η CML προσφέρει αυτό το ενδιάμεσο επίπεδο. Οι στόχοι της είναι να είναι υψηλού επιπέδου, πλήρης, επεκτάσιμη, συγχρονισμένη και να υποστηρίζει δράσεις με παραμέτρους.

2.4.3 PAR: Parametrized Action Representation [16]

Η PAR μετατρέπει τη φυσική γλώσσα σε εντολές για εικονικούς πράκτορες και κάνει την επικοινωνία μεταξύ χρήστη και πράκτορα να μοιάζει με επικοινωνία μεταξύ ζωντανών ανθρώπων.

2.4.4 STEP: Scripted Technology for Embodied Persona [17]

Χρησιμοποιείται για πράξεις επικοινωνίας όπως χειρονομίες και στάσεις σώματος και μπορεί να επεκταθεί για ομιλία και εκφράσεις προσώπου. Βασίζεται στο πρότυπο H-ANIM και έχει υλοποιηθεί με τη γλώσσα DLP, μια γλώσσα λογικού προγραμματισμού. Μετά τη STEP δημιουργήθηκε και μια έκδοση σε XML, η XSTEP[18].

2.4.5 BML: Behavior Markup Language [19][20]

Η BML είναι αποτέλεσμα μιας διεθνούς προσπάθειας για τη δημιουργία ενός ενοποιημένου πλαισίου για την παραγωγή συμπεριφοράς για εικονικούς πράκτορες που έχουν την ικανότητα συζήτησης. Η ομάδα υπεύθυνη για τη BML προτείνει ένα μοντέλο που περιέχει τρία επίπεδα και ονομάζεται SAIBA (Situation, Agent, Intention, Behavior, Animation). Τα στάδια αυτά είναι ο σχεδιασμός της πρόθεσης, ο σχεδιασμός της συμπεριφοράς και η πραγματοποίηση της συμπεριφοράς. Η FML, που περιγράφει πρόθεση χωρίς υλική συμπεριφορά ενώνει τα δύο πρώτα επίπεδα ενώ η BML που περιγράφει την επιθυμητή υλική συμπεριφορά ενώνει το δεύτερο στάδιο με το τρίτο. Η BML είναι επέκταση και βελτίωση προηγούμενων scripting languages.

2.4.6 VHML: Virtual Human Markup Language [21]

Μια προσπάθεια για την περιγραφή όλων των διαφορετικών στοιχείων της συμπεριφοράς ενός αληθοφανούς εικονικού πράκτορα. Αποτελείται από τα ακόλουθα υποσυστήματα

- *DMML*: Dialogue Managing Markup Language
- *FAML*: Facial Animation Markup Language
- *BAML*: Body Animation Markup Language
- *SML*: Speech Markup Language
- *EML*: Emotion Markup Language
- *GML*: Gesture Markup Language

Αρκετά σημεία της VHML δεν έχουν οριστεί πλήρως ακόμα. Το BAML προς το παρόν κληρονομεί τα στοιχεία των EML και GML. Θα επιτρέπει τον ορισμό κίνησης όχι μόνο σε υψηλό επίπεδο, με την ύπαρξη κάποιων έτοιμων κινήσεων, αλλά και σε χαμηλό επίπεδο με βάση το H-ANIM.

2.4.7 Σύγκριση χαρακτηριστικών των γλωσσών συμπεριφοράς [22]

Παρακάτω παρουσιάζονται κάποιες ομοιότητες και διαφορές μεταξύ των γλωσσών συμπεριφοράς που αναφέρθηκαν.

	Scripting languages		Both Scripting and Representation Languages		Representation Languages	
	AML	STEP	CML	VHML	PAR	BML
Approach						
Objectives	Animation	Animation	Animation	Animation	Parametrised action	Description of multimodal behavior
Format						
XML	✓		✓	✓		✓
Specification Elements						
Character Definition			✓	✓	✓	
Animation	✓	✓	✓	✓	✓	
Dialogue Acts				✓	✓	✓
World		✓			✓	
Actions/Behaviour	✓	✓	✓	✓	✓	✓
Voice Controls				✓		✓
Animation Control						
Inhibiting animation	✓					✓
Merging	✓	✓	✓	✓	✓	
Synchronisation	✓	✓	✓	✓	✓	✓
Additional Parameters	✓	✓	✓	✓	✓	✓
Parametrised actions	✓	✓	✓	✓	✓	✓

Feedback to application		✓		✓	✓	✓
Specification Granularity						
Extensibility	✓	✓	✓	✓	✓	✓
Macro Elements	✓	✓	✓	✓	✓	✓
Micro Elements		✓	✓	✓		✓
Believability Attributes						
Emotions	✓		✓	✓	✓	
Personality			✓	✓	✓	
Character Type						
Human Like	✓	✓	✓	✓	✓	✓
Non-Human			✓	✓	✓	
Character Parts/Modules						
Face	✓		✓	✓	✓	✓
Body	✓	✓	✓	✓	✓	✓
Speech	✓		✓	✓	✓	✓

2.5 Real-time Character Engines [23]

Όλα τα παραπάνω συνδυάζονται σε μηχανές σχεδιασμού εικονικών χαρακτήρων. Αυτό το είδος λογισμικού επιτρέπει την αναπαραγωγή και προβολή (rendering) ενός εικονικού χαρακτήρα και την αυτόματη παραγωγή κίνησης με βάση τη συμπεριφορά του πράκτορα. Παρουσιάζονται τα αποτελέσματα μιας ανάλυσης και σύγκρισης τέτοιων λογισμικών με βάση τέσσερις βασικές λειτουργίες και την προσθήκη σε αυτά του REVEWorlds.

	Appearance	Rigs	Animation Generators	Control
Cal3d	A basic polygon renderer	Skeleton and morph targets	Animation data and basic blending	N/A

Piavca	As Cal3d	As Cal3d + hardware morph targets	Generic motion generation and combining system	Event based and continuous control methods
HALCA	As Cal3d + pluggable shader architecture	As Cal3d + dual quaternion skinning	Numerous data driven and procedural generators	Event based, real-time tracking and physics based interaction
UNREAL	Fully featured real-time rendering	Skeletons, morphs	Data driven, Procedural, Inverse Kinematics, Blend Controllers, Physics Controllers	Custom AI using UnrealScript, Visual Scripting
Jack	A basic polygon renderer	Skeleton	Numerous procedural generators	Various, including AI and speech driven interaction
VHD++	Fully featured renderer including hair and cloth	Skeleton, facial bones and morph targets	Data driven and procedural generators + real time tracking	Real-time input and AI
Havok	N/A	Skeleton and morph targets	Extensive data driven, inverse kinematics and physics based generators	N/A
Natural Motion	N/A	Skeleton	Extensive data driven and physics based generators	N/A

SAIBA+ GRETA	Polygon renderer	Complex facial rig	Complex procedural controllers tailored to non-verbal communication	Via the BML language
REVEWorlds	Xj3d	H-ANIM LOA-1	Custom code	Custom code

ΚΕΦΑΛΑΙΟ 3

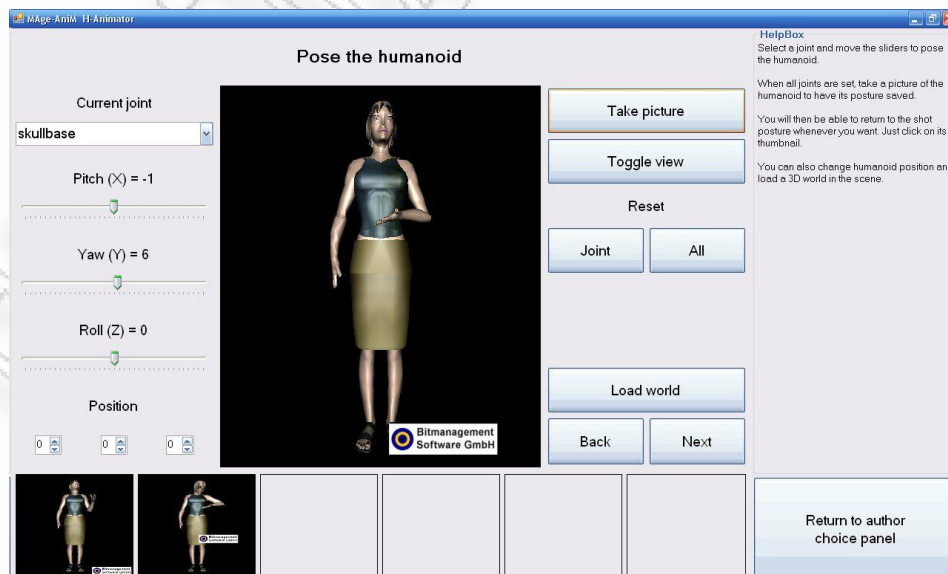
Ανάλυση

3.1 Το πρότυπο H-ANIM

Για την κατασκευή της κίνησης χρησιμοποιήθηκε το πρότυπο H-ANIM, το οποίο είναι ένα πρότυπο περιγραφής σκελετικών δομών. Έχει σχεδιαστεί κυρίως για να αναπαριστά ανθρωπόμορφους χαρακτήρες αλλά δε θέτει περιορισμούς στον αριθμό των άκρων (χέρια, πόδια, κεφάλια) και μπορεί να χρησιμοποιηθεί και για ζώα ή φανταστικούς χαρακτήρες.

3.2 Πλεονεκτήματα H-ANIM

Το H-ANIM υποστηρίζεται από αρκετά εργαλεία μοντελοποίησης (open source και εμπορικά, για παράδειγμα Seamless3d, Flux, VizX3d, H-Animator) και μπορεί να αποθηκευτεί σε μορφή X3D ή VRML. Τα αρχεία αυτά δε χρειάζονται κάποιο εξειδικευμένο software, αλλά μπορούν να προβληθούν σε κάποιο VRML ή X3D browser(παράδειγμα τέτοιων browser είναι το Xj3d, το οποίο χρησιμοποιείται από το εικονικό περιβάλλον REVE Worlds για την αναπαράσταση γραφικών και το Instant Player).



Εικόνα 4:H-ANIMATOR [42]

Η επαναχρησιμοποίηση των animations από μοντέλα που έχουν τον ίδιο αριθμό αρθρώσεων είναι εύκολη (αν και μπορεί να υπάρχουν διαφορές μεταξύ μοντέλων που διαφέρουν σημαντικά στη γεωμετρία τους) γιατί το πρότυπο ορίζει συγκεκριμένη ιεραρχία και ονόματα για τις αρθρώσεις και συγκεκριμένη αρχική στάση σώματος.

3.3 Παραδείγματα χρήσεων του H-ANIM

Το H-ANIM μπορεί να χρησιμοποιηθεί σε σχεδόν οποιαδήποτε τρισδιάστατη εφαρμογή που απαιτεί την αναπαράσταση ανθρωπόμορφων μοντέλων. Κάποια παραδείγματα χρήσης του H-ANIM είναι σε εικονικό εκπαιδευτικό περιβάλλον [24], εικονικό κατάστημα ρούχων [25], ευφυή εικονική αφήγηση [26], τον εικονικό εκπαιδευτή Jacob [27] και την επίδειξη της ελληνικής νοηματικής γλώσσας για εκπαιδευτικούς σκοπούς[28]. Άλλα είδη εφαρμογών που ωφελούνται από το H-ANIM είναι τα παιχνίδια[32] και οι εργονομικές μελέτες [29].

3.4 Γεωμετρία H-ANIM σώματος [5]

Η γεωμετρία που περιγράφει ένα H-ANIM σώμα μπορεί να είναι σκελετική ή δερματική.

Όταν χρησιμοποιείται η σκελετική μέθοδος, τότε το σώμα αποτελείται από ξεχωριστά γεωμετρικά κομμάτια (segments) τα οποία ενώνονται μεταξύ τους με joints. Αυτός ο τρόπος αναπαράστασης είναι υπολογιστικά αποδοτικός, αλλά πολλές φορές δημιουργούνται ασυνέχειες στο μοντέλο με αποτέλεσμα να μειώνεται η αληθοφάνεια του μοντέλου.

Η δερματική μέθοδος περιγράφει το σώμα ως ένα ενιαίο τμήμα γεωμετρίας που καλύπτεται από δέρμα. Κάθε φορά που μια άρθρωση εκτελεί περιστροφή, τότε παραμορφώνεται ανάλογα το δέρμα. Έτσι, δε δημιουργούνται κενά τμήματα μεταξύ των μερών του σώματος.

3.4.1 Κόμβοι του H-ANIM

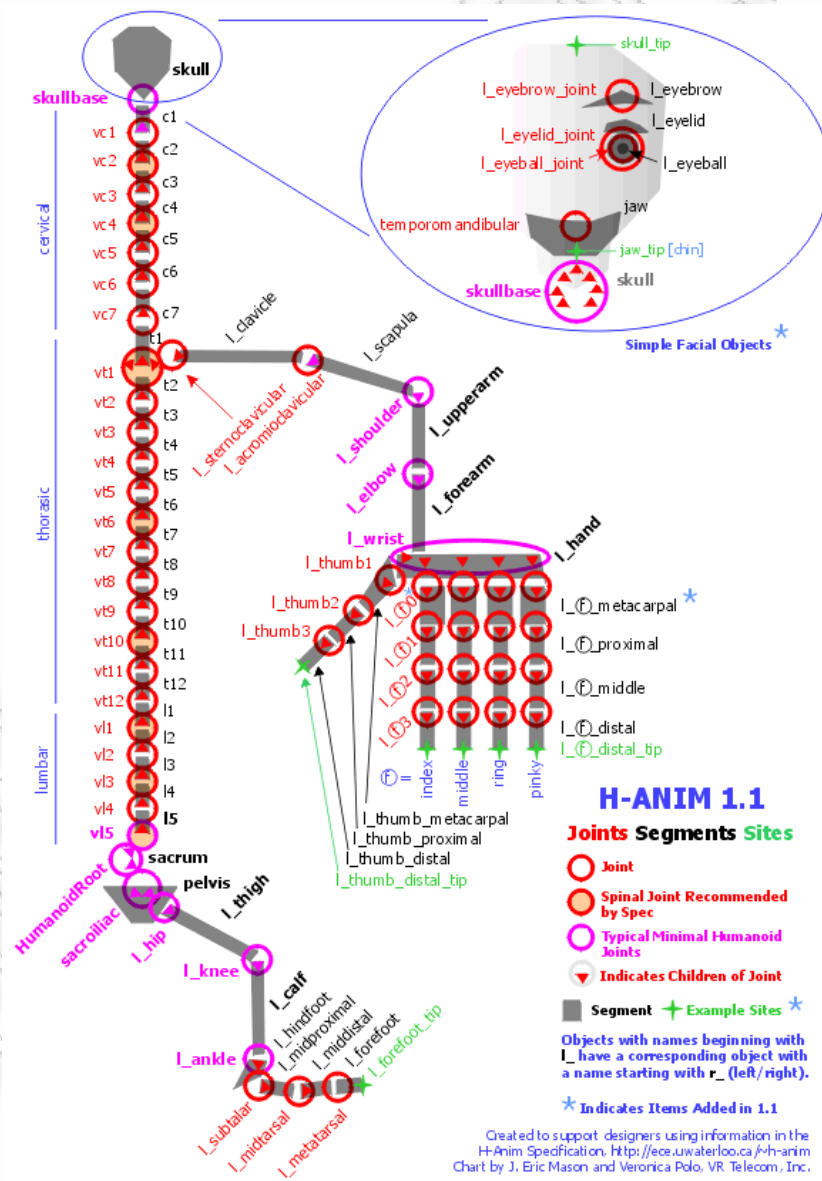
Ένας εικονικός χαρακτήρας που αναπαρίσταται σε H-ANIM αποτελείται από τους παρακάτω κόμβους X3d .

- *Humanoid* : Είναι ο κεντρικός κόμβος που περιλαμβάνει τους υπόλοιπους. Η γεωμετρία του χαρακτήρα μπορεί να είναι είτε σκελετική και να αποτελείται από segments, είτε να είναι ένα συνεχές τμήμα και να καλύπτεται από δέρμα
- *Joint* : Συμβολίζει τις αρθρώσεις του σώματος και χειρίζεται την κίνηση.
- *Segment* : Ορίζει τα τμήματα του σώματος, όπως χέρια, πόδια κτλ
- *Site* : Χρησιμοποιείται για να ορίσει άκρα που μπορούν να χρησιμοποιηθούν από συστήματα inverse kinematics, σημεία συγκόλλησης για αξεσουάρ, όπως κοσμήματα και ρούχα και για να ορίσει θέσεις για viewports.
- *Displacer* : Ορίζει το εύρος της κίνησης του αντικειμένου στο οποίο είναι ενσωματωμένο

3.4.2 Levels of Articulation [29]

Το πρότυπο ορίζει όλες τις αρθρώσεις του ανθρώπινου σώματος αλλά δεν απαιτούν όλες οι εφαρμογές το μεγαλύτερο βαθμό λεπτομέρειας και αληθοφάνειας της κίνησης. Έτσι, το H-ANIM προσφέρει τέσσερα επίπεδα αριθμού αρθρώσεων (Levels of Articulation).

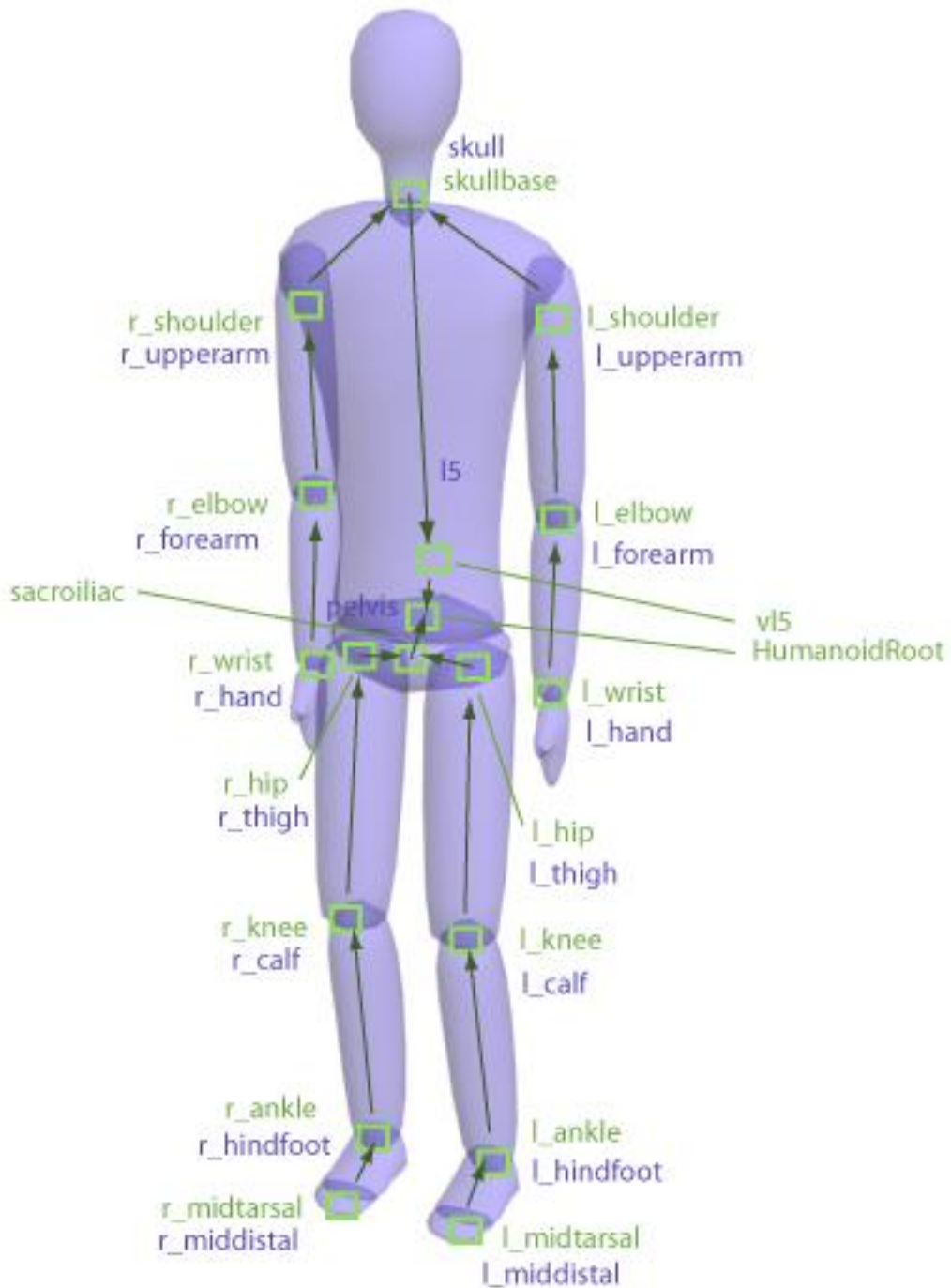
- *LOA-0*: Ορίζεται μόνο ο κεντρικός κόμβος του μοντέλου (οι υπόλοιπες αρθρώσεις παίρνουν προκαθορισμένες τιμές).
- *LOA-1*: Ορίζονται οι 18 κύριες αρθρώσεις του σώματος (χωρίς τη σπονδυλική στήλη και τις αρθρώσεις των χεριών)
- *LOA-2*: Ορίζονται 71 αρθρώσεις, οι οποίες περιλαμβάνουν τις αρθρώσεις των χεριών με μια απλή σπονδυλική στήλη
- *LOA-3*: ορίζεται η πλήρης ιεραρχία του H-ANIM που αποτελείται από 89 αρθρώσεις .



Εικόνα 5: Όλες οι αρθρώσεις που προσφέρει το H-ANIM[42]

Ένα animation που έχει δημιουργηθεί για κάποιο LOA μπορεί να χρησιμοποιηθεί από οποιοδήποτε μοντέλο έχει το ίδιο ή ανώτερο LOA.

Τα animations που έχουν δημιουργηθεί για αυτή τη διατριβή ακολουθούν το LOA-1.



Εικόνα 6: Οι αρθρώσεις του LOA-1 και οι θέσεις τους σε ένα ανθρωπόμορφο μοντέλο[41]

ΚΕΦΑΛΑΙΟ 4

Σχεδιασμός

4.1 Εισαγωγή

Σκοπός αυτής της διπλωματικής είναι η ενσωμάτωση κίνησης στους εικονικούς πράκτορες του ReveWorlds. Ο σχεδιασμός έγινε έτσι, ώστε τα animations και τα μοντέλα να είναι ανεξάρτητα μεταξύ τους αλλά να ακολουθούν ένα συγκεκριμένο πρότυπο. Έτσι είναι εξασφαλισμένη η συμβατότητα τους και δε χρειάζεται να δημιουργούνται για κάθε καινούριο μοντέλο νέα animations ή το αντίστροφο. Ο πιο απλός τρόπος να επιτευχθεί αυτό είναι όλα τα μοντέλα να έχουν τον ίδιο αριθμό αρθρώσεων

4.2 Η πλατφόρμα REVE και το REVEWorlds [33]

Η αναπαράσταση REVE αποτελεί μία νέα προσέγγιση για την αντιμετώπιση των εγγενών δυσχερειών που δημιουργεί η πολυμορφία των αναγκών του χώρου, πάνω σε μία συγκεκριμένη θεωρητική βάση, με σαφή έμφαση στην υποστήριξη ευφύων εικονικών πρακτόρων και με στόχο τη δυνατότητα άμεσης και συστηματικής εφαρμογής σε πλήθος εφαρμογών, την ευελιξία και την επεκτασιμότητα.[34]

Επιπλέον το REVE προσεγγίζει σε σχεδιαστικό επίπεδο, βασικές έννοιες όπως τον κόσμο και το αντικείμενο, επιτρέποντας έτσι ετερογενείς λύσεις μοντελοποίησης, όπου το κάθε τμήμα εξυπηρετεί ένα συγκεκριμένο σκοπό ενώ παράλληλα συνεργάζεται με τα υπόλοιπα τμήματα με διαφανή και πρότυπο τρόπο. Για παράδειγμα, η φυσική αναπαράσταση ενός εικονικού αντικειμένου μπορεί να υλοποιηθεί με ένα γράφημα σκηνής ενώ η σημασιολογία με μία αναπαράσταση συμβολικής γνώσης. Τα δύο μέρη του ολοκληρωμένου μοντέλου έχουν πρόσβαση το ένα στο άλλο, καθιστώντας δυνατή την έκφραση φυσικών στοιχείων του εικονικού αντικειμένου μέσω της σημασιολογίας του, καθώς και την τροποποίηση της σημασιολογίας μέσω της λειτουργικότητας και της αλληλεπιδραστικότητας της φυσικής αναπαράστασης του μοντέλου. Το αποτέλεσμα είναι ένα πολυδιάστατο αλλά σταθερό μοντέλο της ίδιας οντότητας.

Το REVE Worlds είναι μια μερική υλοποίηση ενός ευφυούς εικονικού περιβάλλοντος που βασίζεται στο REVE. Είναι υλοποιημένο σε Java και χρησιμοποιεί το Xj3d για την προβολή περιεχομένου VRML και X3D. Χειρίζεται εικονικούς κόσμους, που αρχικοποιεί με δεδομένα από

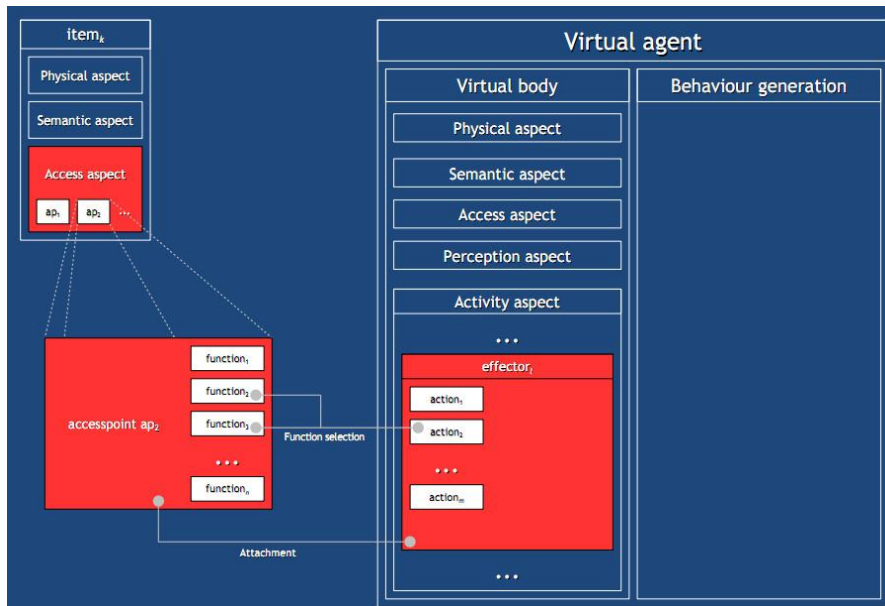
ορισμούς VERL και εικονικά σώματα. Υποστηρίζει πλήθος μεθόδων για τον έλεγχο τους οι οποίες βασίζονται στο AIP, ένα πρωτόκολλο για τη σύνδεση εικονικών σωμάτων με components για την παραγωγή συμπεριφοράς και νόησης (cognition). Αυτά περιλαμβάνουν ένα CLI-based interface κατάλληλο για τον απευθείας πειραματισμό, και έλεγχο των εικονικών σωμάτων.

Έχει αναπτυχθεί ένας αριθμός εφαρμογών πρακτόρων για εικονικά περιβάλλοντα REVE. Ο σκοπός του είναι να ερευνηθεί η δυνατότητα του REVE να υποστηρίξει εφαρμογές ευφυών εικονικών περιβαλλόντων. Συγκεκριμένα, γίνεται προσπάθεια στην αξιολόγηση της αποτελεσματικότητας του REVE στην αντιμετώπιση των αναγκών που έχουν οι πράκτορες σε εικονικά περιβάλλοντα, όπως pathfinding σε πραγματικό χρόνο, την αλληλεπίδραση με άλλα αντικείμενα και άλλους πράκτορες, την αντίληψη του κόσμου και τη δημιουργία νοητικών μοντέλων από αυτά, την παραγωγή συμπεριφοράς, την εκδήλωση σκοπού (σκοπιμότητας), χωρικού συλλογισμού (spatial reasoning). Σε πιο τεχνικό επίπεδο, γίνεται έλεγχος του συστήματος όταν υπάρχουν πολλοί πράκτορες που συνυπάρχουν και υπολογίζεται ο βαθμός της φορητότητας των ορισμών των πρακτόρων και των υλοποιήσεών τους σε διαφορετικά εικονικά περιβάλλοντα.

4.2.1 Εικονικοί πράκτορες στο REVEWorlds[35]

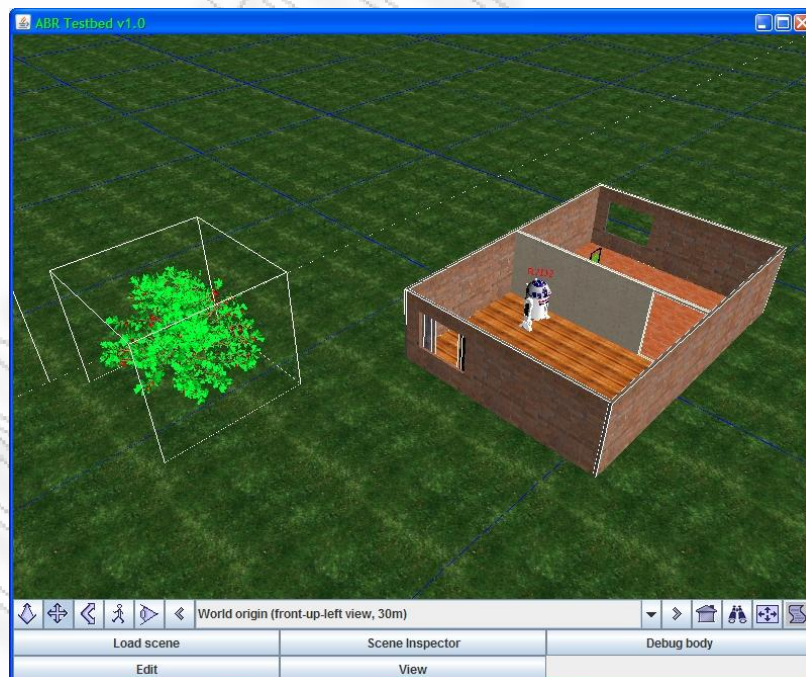
Ένας εικονικός χαρακτήρας στο REVEWorlds είναι το σύνολο των παρακάτω aspects

- *Physical Aspect*: παρουσίαση εικονικού αντικειμένου
- *Semantic Aspect*: πληροφορία διαθέσιμη προς αντίληψη
- *Access Aspect*: λειτουργικότητα εικονικού αντικειμένου
- *Perception Aspect*: αντίληψη εικονικών αντικειμένων
- *Activity Aspect*: δράση επί εικονικών αντικειμένων



Εικόνα 7: Μοντέλο δράσης εικονικού πράκτορα[42]

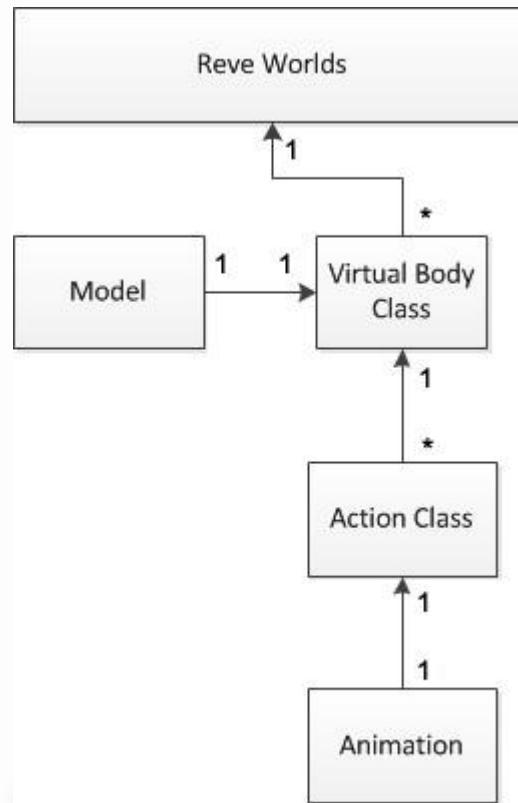
Το κάθε aspect υποστηρίζει συγκεκριμένες λειτουργίες του εικονικού περιβάλλοντος. Το aspect που αφορά την εμφάνιση του πράκτορα είναι το physical aspect και τα aspects που αφορούν το animation είναι τα perception, access και activity aspects. Άρα οι κλάσεις που υλοποιούν το σώμα και την κίνηση πρέπει να διαβάζουν τα εξωτερικά δεδομένα και να τα αντιστοιχούν στις λειτουργίες των aspects.



Εικόνα 8: Το REVEWorlds και ένας εικονικός πράκτορας.[42]

4.3 Δομή

Τα δεδομένα των μοντέλων και των animation περιέχονται σε εξωτερικά αρχεία για τα οποία απαιτούνται ενδιάμεσες κλάσεις που παρέχουν το κατάλληλο interface για τη σύνδεση με το REVEWorlds.



Εικόνα 9: Η δομή για την ενσωμάτωση κίνησης

Το αρχείο του μοντέλου περιέχει πληροφορίες σχετικά με τη γεωμετρία (εμφάνιση) και τις αρθρώσεις του μοντέλου. Όταν υπάρχουν μεγάλες αποκλίσεις στις γεωμετρικές των μοντέλων μπορεί να υπάρχει κάποια απόκλιση και στο animation αλλά στις περισσότερες περιπτώσεις δε θα δημιουργούνται σημαντικά προβλήματα.

Το αρχείο του animation περιγράφει τις κινήσεις των αρθρώσεων που απαρτίζουν ένα animation. Το animation ολοκληρώνεται σε ένα κύκλο και σε κατάλληλες χρονικές στιγμές του κύκλου ορίζονται περιστροφές των αρθρώσεων.

Το μοντέλο και το animation συνδέονται με το Reve μέσω των κλάσεων Virtual Body Class και Action Class αντίστοιχα. Το Reve Worlds μπορεί να περιέχει πολλά σώματα (Virtual Body) και κάθε σώμα μπορεί να περιέχει πολλές δράσεις (actions). Η επιλογή του μοντέλου και των animations, όπως και ο έλεγχος τους γίνεται μέσω του ReveWorlds.

ΚΕΦΑΛΑΙΟ 5

Υλοποίηση

5.1 Εισαγωγή

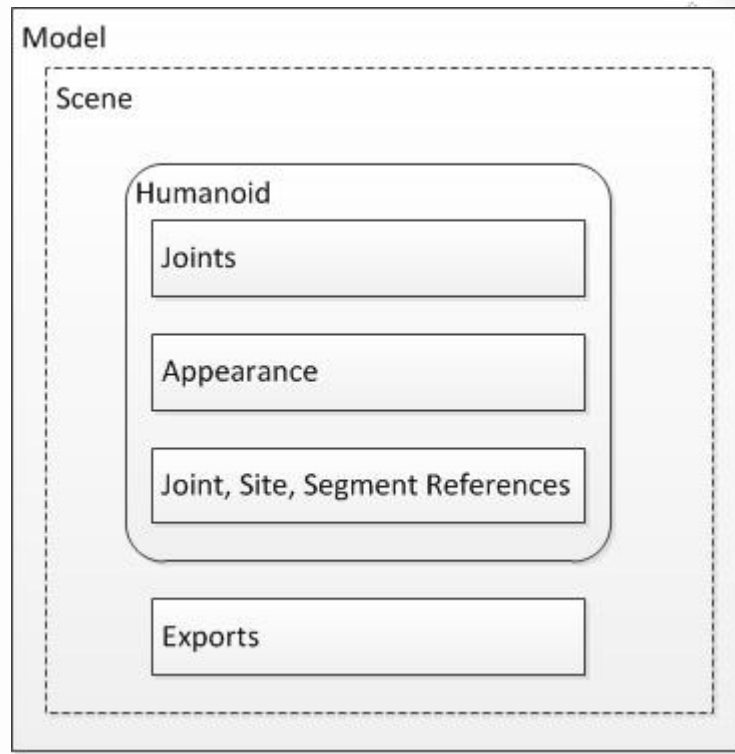
Η τελική υλοποίηση είχε ως βασική προϋπόθεση την απόλυτη συμβατότητα με το REVE Worlds. Η περιγραφή του H-ANIM μοντέλου και των animations έγινε σε X3d για να είναι δυνατή η απόδοσή τους από τον X3d viewer που χρησιμοποιεί το REVE Worlds για την αναπαράσταση γραφικών. Για να συνδεθούν τα αρχεία X3d με το REVE Worlds, το οποίο είναι υλοποιημένο σε java, παραχωρήθηκαν από τον δημιουργό του REVE Worlds κ.Αναστασάκη κάποιες κλάσεις java οι οποίες επεκτάθηκαν κατάλληλα.

5.1.1 Σύντομη περιγραφή του X3D[30][31]

Το X3d είναι ο διάδοχος της VRML και είναι ένα πρότυπο ISO για τρισδιάστατα διαδικτυακά γραφικά που βασίζεται στην XML. Προσφέρει ένα σύστημα για την αποθήκευση, την ανάκτηση και την αναπαραγωγή γραφικού περιεχομένου σε πραγματικό χρόνο, το οποίο υλοποιείται με μια ανοιχτή αρχιτεκτονική που υποστηρίζει μεγάλο εύρος εφαρμογών. Το X3d μπορεί να αναπαραστήσει πολυγωνική γεωμετρία, παραμετρική γεωμετρία, ιεραρχικούς μετασχηματισμούς, φωτισμό, υλικά κτλ όπως και δισδιάστατα γραφικά.

5.2 Αρχεία X3d

5.2.1 Περιγραφή αρχείου X3d για το σώμα



Εικόνα 10: Η δομή του X3d αρχείου για το μοντέλο

Το αρχείο του μοντέλου αποτελείται από το μοντέλο και τα exports του και συνάδει με τις προδιαγραφές του H-ANIM LOA-1, δηλαδή περιέχει τα παρακάτω 18 joints με τις αντίστοιχες default θέσεις και αναφορές στα Joints για να χρησιμοποιηθούν από τις συναρτήσεις για την κίνηση.

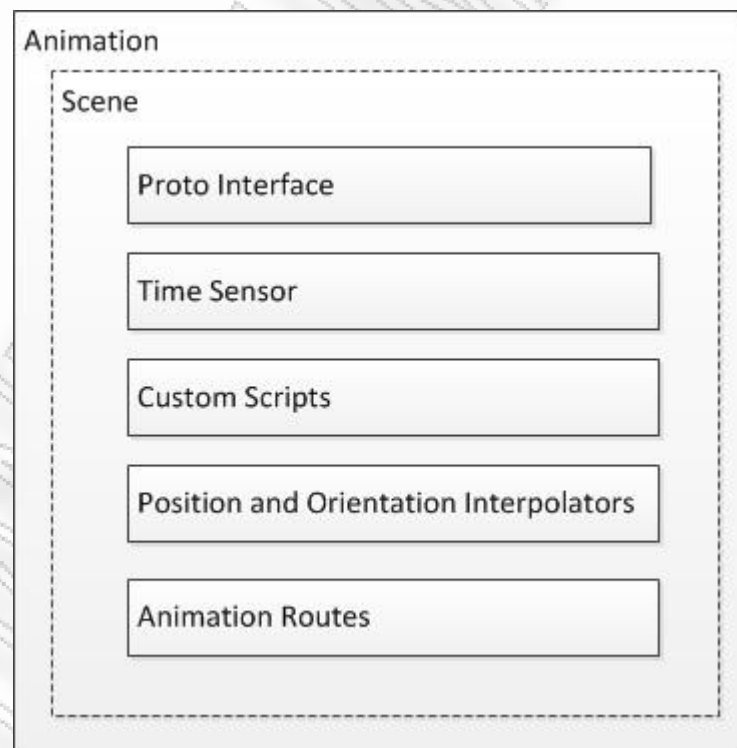
Joint	Default center	Joint object	Default center
HumanoidR	0.0000 0.8240	r_midtarsal	-0.0801 0.0000
sacroiliac	0.0000 0.9149	v15	0.0028 1.0568 -
l_hip	0.0961 0.9124 -	skullbase	0.0044 1.6209
l_knee	0.1040 0.4867	l_shoulder	0.2029 1.4376 -
l_ankle	0.1101 0.0656 -	l_elbow	0.2014 1.1357 -

l_midtarsal	0.1086 0.0001	l_wrist	0.1984 0.8663 -
r_hip	-0.0950 0.9171	r_shoulder	-0.1907 1.4407 -
r_knee	-0.0867 0.4913	r_elbow	-0.1949 1.1388 -
r_ankle	-0.0801 0.0712 -	r_wrist	-0.1959 0.8694 -

Δεν είναι υποχρεωτικό να υπάρχουν όλα τα joints. Το μόνο απαραίτητο joint, σύμφωνα με το H-ANIM είναι το HumanoidRoot. Στην περίπτωση που υπάρχουν λιγότερα joints, το μοντέλο θα μπορεί να υποστηρίξει μόνο τα animations που έχουν τα συγκεκριμένα (ή λιγότερα) joints. Είναι προτιμότερο όλα τα σώματα να έχουν και τα 18 joints για να είναι ευκολότερη η επαναχρησιμοποίησή τους χωρίς προβλήματα.

Οι κόμβοι HAnimSegments αναπαριστούν τη γεωμετρία του σώματος (τμήματα χεριών, ποδιών κτλ). Αυτά τα τμήματα χρησιμεύουν στην οπτικοποίηση του σώματος.

5.2.2 Περιγραφή αρχείου X3d για το animation



Εικόνα 11: Η δομή του αρχείου X3d για το animation

Η κίνηση προσδιορίζεται σε ένα proto που περιέχει PositionInterpolators και OrientationInterpolators τα οποία ορίζουν τις νέες θέσεις και προσανατολισμούς των joints του σώματος και περιέχει ένα TimeSensor για τον έλεγχο της κίνησης (ταχύτητα, έναρξη και παύση). Τα Routes χρησιμοποιούνται για να ανταλλάζουν μεταβλητές μεταξύ κόμβων. Στη συγκεκριμένη περίπτωση, χρονικές στιγμές από το TimeSensor στα Interpolators.

5.2.3 Παράδειγμα Route

```
<ROUTE fromField='fraction_changed' fromNode='TIMER' toField='set_fraction'
toNode='R_SHOULDER_ANIMATOR' />
```

Καθώς περνάει ο χρόνος, η τιμή του στέλνεται από τον κόμβο TIMER στους animators.

5.2.4 Παράδειγμα Animator

```
<PositionInterpolator DEF='HUMANOIDROOT_POSITION_ANIMATOR' key='0 0.04167
0.125 0.1667 0.2083 0.25 0.2917 0.375 0.4583 0.5 0.5417 0.5833 0.625 0.7083
0.75 0.7917 0.875 0.9167 1' keyValue='0 -0.00928 0 0 -0.003858 0 0 -0.008847 0
0 -0.01486 0 0 -0.02641 0 0 -0.03934 0 0 -0.0502 0 0 -0.07469 0 0 -0.02732 0 0
-0.01608 0 0 -0.01129 0 0 -0.005819 0 0 -0.002004 0 0 -0.002579 0 0 -0.0143 0
0 -0.03799 0 0 -0.05648 0 0 -0.045 0 0 -0.00928 0'>

<IS>

<connectnodeField='value_changed'protoField='HumanoidRoot_translation_changed' />
</IS>
</PositionInterpolator>
```

Το πεδίο key του PositionInterpolator ορίζει χρονικές στιγμές από την έναρξη ως το τέλος ενός κύκλου και αντίστοιχες θέσεις του joint σε κάθε χρονική στιγμή. Το πεδίο keyValue περιέχει τόσες τριάδες τιμών όσες το πεδίο key.

Το OrientationInterpolator λειτουργεί με παρόμοιο τρόπο. Για κάθε χρονική στιγμή ορίζεται μια τετράδα τιμών. Οι πρώτες τρεις τιμές είναι ο άξονας περιστροφής και η τέταρτη τιμή είναι η γωνία περιστροφής.

Ένα animator μπορεί να περιέχει και κόμβους script, με κώδικα δημιουργημένο από το χρήστη αν χρειάζεται να υπολογίσει τις τιμές κάποιων μεταβλητών, για παράδειγμα την ταχύτητα ή κάποια περιστροφή. Τα scripts γράφονται σε ECMAScript.

5.2.5 Script για την κίνηση του χεριού

Με αυτό το script υπολογίζονται οι γωνίες του αγκώνα και του ώμου του δεξιού χεριού με βάση το τελικό σημείο που πρέπει να φτάσει η άκρη του χεριού. Όταν βρεθούν οι γωνίες δίνονται σαν έξοδοι οι περιστροφές του αγκώνα και του ώμου

```
<![CDATA[
javascript:
function input(val, time) {
var el_ang;
var sh_ang;
  if(val[1]<=-0.35){
el_ang=0.8+((0.7+val[1])*2);
sh_ang=0.51;
  }
  else if(val[1] > -0.35)
  {
el_ang=1.3;
sh_ang=0.9+(0.7+(val[1]*2.8));
  }
  elbow_output =new MFRotation(new SFRotation(0, 0, 1, 0),new SFRotation(-1,
0, 0, el_ang),new SFRotation(-1, 0, 0,el_ang));
  shoulder_output = new MFRotation(new SFRotation(-0.9999999999999996, 0, 0,
0.20943951023932),new SFRotation(-1, 0, 0, 0.314159265358979),new SFRotation(-
1, 0, 0, sh_ang));
  }
  ]]>
</Script>
```

5.3 Αρχεία Java

5.3.1 Περιγραφή της κλάσης HanimBody[35]

Αυτή η κλάση δημιουργεί ένα εικονικό σώμα από ένα H-ANIM LOA-1 σώμα για το ReveWorlds και επιτρέπει σε actions να έχουν πρόσβαση σε LOA1 κόμβους τους οποίους μπορούν, έτσι, να κινούν με κατάλληλα animations.

Το πρώτο πράγμα που κάνει η κλάση είναι να εντοπίσει τους 18 κόμβους του LOA-1. Σαν sensors και effectors χρησιμοποιούνται τα H-ANIM sites sellion, r_dactylion και l_dactylion. Οι

sensors είναι η αναπαράσταση του αισθητηρίου και αναλαμβάνουν τη λήψη, επεξεργασία και αποθήκευση αισθητηρίων δεδομένων. Ο Effector είναι η αναπαράσταση του ενσώματου μέσου εκτέλεσης ενεργειών. Τα στάδια λειτουργίας του είναι η συσχέτιση με accesspoint, η επιλογή των functions(τους μηχανισμούς, δηλαδή, που μεταβάλουν το πληροφοριακό περιεχόμενο κάποιου αντικειμένου) και η εκτέλεση των actions, τα οποία είναι οι μηχανισμοί εκτέλεσης functions βάσει συγκεκριμένου προτύπου. Είναι μέρος του activity aspect και συνδέεται μέσω κάποιου accesspoint με το access aspect κάποιου αντικειμένου. Οι effectors και sensors ενσωματώνονται στο εικονικό σώμα με τον συσχετισμό τους με κάποιο κόμβο του physical aspect.

```

if (!preprocessHAnimNode("hanim_HumanoidRoot", hanimNodes)) {
    throw new IllegalArgumentException("Node \"HumanoidRoot\"
not found");
}
preprocessHAnimNode("hanim_sacroiliac", hanimNodes);
preprocessHAnimNode("hanim_l_hip", hanimNodes);
preprocessHAnimNode("hanim_l_knee", hanimNodes);
preprocessHAnimNode("hanim_l_ankle", hanimNodes);
preprocessHAnimNode("hanim_l_midtarsal", hanimNodes);
preprocessHAnimNode("hanim_r_hip", hanimNodes);
preprocessHAnimNode("hanim_r_knee", hanimNodes);
preprocessHAnimNode("hanim_r_ankle", hanimNodes);
preprocessHAnimNode("hanim_r_midtarsal", hanimNodes);
preprocessHAnimNode("hanim_vl5", hanimNodes);
preprocessHAnimNode("hanim_skullbase", hanimNodes);
preprocessHAnimNode("hanim_l_shoulder", hanimNodes);
preprocessHAnimNode("hanim_l_elbow", hanimNodes);
preprocessHAnimNode("hanim_l_wrist", hanimNodes);
preprocessHAnimNode("hanim_r_shoulder", hanimNodes);
preprocessHAnimNode("hanim_r_elbow", hanimNodes);
preprocessHAnimNode("hanim_r_wrist", hanimNodes);
preprocessHAnimNode("hanim_sellion", hanimNodes); //sensor
preprocessHAnimNode("hanim_l_dactylion", hanimNodes); //effector
preprocessHAnimNode("hanim_r_dactylion", hanimNodes); //effector

```

Ο μόνος κόμβος που είναι απαραίτητος από τους παραπάνω για να λειτουργήσει ένα σώμα είναι ο HumanoidRoot. Τα animations για τους κόμβους που δεν είναι διαθέσιμοι απλά δεν θα οριστούν. Αν δε βρεθούν οι sellion, r_dactulion ή l_dactylion, τότε χρησιμοποιείται ο HumanoidRoot σαν sensor και hand effector.

Στην κλάση υπάρχουν και άλλες πληροφορίες όπως viewpoints κτλ, ενώ στο τέλος γίνεται προσθήκη του sensor

```

X3DNode sensorNode = sellionNode != null ? sellionNode : humanoidRootNode;
OrthogonalViewFrustumSensor sensor = new OrthogonalViewFrustumSensor("sensor",
sensorNode, null, null);

    sensor.setScene(scene);

    sensor.create();

    this.sensors.add(sensor);

```

Για την κίνηση του σώματος δημιουργείται ένα access point με ένα AccessPointTranslateFunctionL1 που επιτρέπει τον μετασχηματισμό του μοντέλου με βάση κάποιο σημείο και την AccessPointRotateFunctionL για την περιστροφή του μοντέλου.

```

SimpleAccessPoint apMotor = new SimpleAccessPoint();

    apMotor.create("motorAccessPoint", humanoidRootNode,
parentsArray);

    AccessPointTranslateFunctionL1 ftMotor = new
AccessPointTranslateFunctionL1();

    .
    .
    .

AccessPointRotateFunctionL1 frMotor = new AccessPointRotateFunctionL1();

```

Το σώμα κινεί ένας effector(motor) , και αν το μοντέλο έχει χέρια δημιουργείται ένας effector για κάθε χέρι. Όταν ένας effector συνδέεται με ένα accesspoint αποκτά πλήρη πρόσβαση στα functions του accesspoint με το οποίο έχει συνδεθεί

Εδώ φαίνεται ένα παράδειγμα προσθήκης κίνησης. Δημιουργείται ένας tMotor που δίνει στο μοντέλο την ικανότητα να προσομοιώνει το ανθρώπινο βάδισμα καθώς κινείται .

```

TranslateActionL1 tMotor = new HAnimBodyMoveActionL1(hanimNodes, scene,
browser);

    tMotor.setScene(scene);

    tMotor.setEffector(efMotor);

    tMotor.create();

```

5.3.2 Περιγραφή της κλάσης για τους animators[35]

Αυτή η κλάση συνδέει τα animations με την κλάση του σώματος για το ReveWorlds. Τα εκκινεί και τα σταματά κατάλληλα και περνάει μεταβλητές που ελέγχουν τη δράση τους (για παράδειγμα την ταχύτητα του animation).

Το κάθε animation είναι ένα action class. Περιγράφει τη λογική εκτέλεση των functions, των ορισμάτων εκτέλεσης και των συμβατών function classes(τα οποία περιγράφουν τα αποτελέσματα και τα ορίσματα εκτέλεσης). Κάθε action class ορίζει ένα σύνολο συμβατών function classes

Τα joints διαβάζονται από την κλάση HanimBody και δημιουργούνται routes για τους animators.

```
scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"HumanoidRoot_translation_changed", humanoidRootNode, "translation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"HumanoidRoot_rotation_changed", humanoidRootNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_hip_rotation_changed", lHipNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_knee_rotation_changed", lKneeNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_ankle_rotation_changed", lAnkleNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_midtarsal_rotation_changed", lMidtarsalNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"r_hip_rotation_changed", rHipNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"r_knee_rotation_changed", rKneeNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"r_ankle_rotation_changed", rAnkleNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"r_midtarsal_rotation_changed", rMidtarsalNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"v15_rotation_changed", v15Node, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"skullbase_rotation_changed", skullbaseNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_shoulder_rotation_changed", lShoulderNode, "rotation");
    scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
"l_elbow_rotation_changed", lElbowNode, "rotation");
```

```

        scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
        "l_wrist_rotation_changed", lWristNode, "rotation");

        scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
        "r_shoulder_rotation_changed", rShoulderNode, "rotation");

        scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
        "r_elbow_rotation_changed", rElbowNode, "rotation");

        scene.getX3DSceneMediator().addRoute(walkAnimatorNode,
        "r_wrist_rotation_changed", rWristNode, "rotation");

```

5.3.3 Παράδειγμα ενέργειας

Εδώ φαίνεται η συνάρτηση που περνάει μια παράμετρο σε μία δράση (συγκεκριμένα ενεργοποιεί το animation)

```

protected void setWalking(boolean isWalking) {
    ((SFBool) walkAnimatorNode.getField("enabled")).setValue(isWalking);
}

```

Οι παρακάτω συναρτήσεις ειδοποιούν για την έναρξη, την ολοκλήρωση ή την αποτυχία της εκτέλεσης μιας ενέργειας.

```

public void executionStarted(ExecutionEvent e) {
    // this is a notification that the function, thus, a single
    execution
    // step, has started executing...
    DebugMessage.printMessage(this, "executionStarted", "Action
    execution started");
    // super.executionStarted(e);
    setCycleInterval((float) (3));
    setWalking(true);
}

// @Override
public void executionCompleted(ExecutionEvent e) {
    // this is a notification that the function, thus, a single
    execution
    // step has completed execution...
    DebugMessage.printMessage(this, "executionCompleted", "Action
    execution completed");
    // super.executionCompleted(e);
}

```

```
        setWalking(false);
    }

    // @Override
    public void executionFailed(ExecutionEvent e) {
        // this is a notification that the function, thus, a single
execution
        // step has failed while executing...
        DebugMessage.printMessage(this, "executionFailed", "Action execution
failed");
        // super.executionFailed(e);
        setWalking(false);
    }
}
```

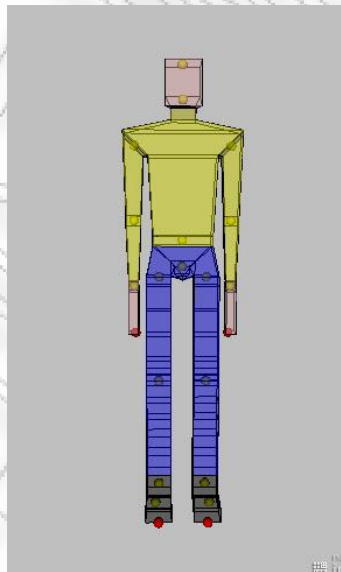

ΚΕΦΑΛΑΙΟ 6

Case Study

6.1 Παραδείγματα λειτουργίας

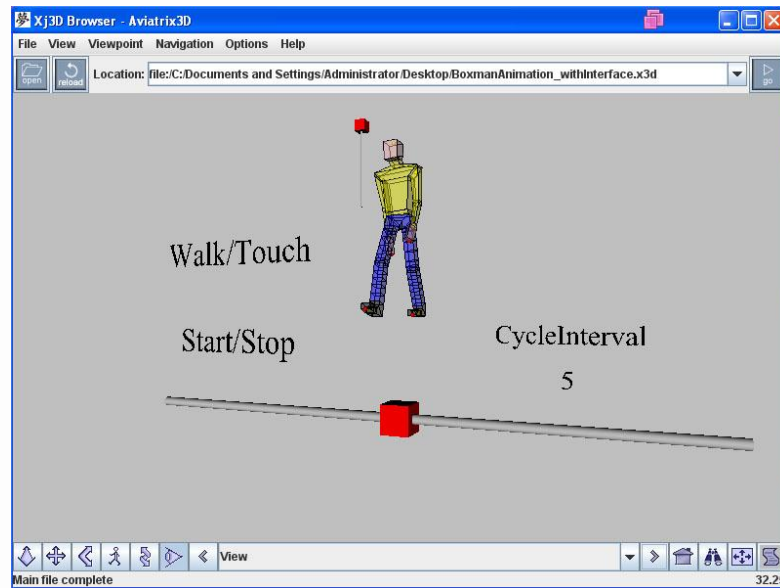
Για την επίδειξη της λειτουργίας του συστήματος χρησιμοποιήθηκε το μοντέλο H-ANIM Boxman [43] και τα animations LOA1_WalkAnimation[44] και TouchAnimation. Τα δύο πρώτα είναι παραδείγματα που παρέχονται από το Web3D για την κατανόηση του X3d και του H-ANIM ενώ το τρίτο δημιουργήθηκε για την διπλωματική με τη χρήση του εργαλείου H-ANIMATOR.

Ο Boxman είναι ένα seamless τρισδιάστατο μοντέλο που αναπαρίστανται μόνο με το δέρμα του. Οι αρθρώσεις και τα sites του έχουν τη μορφή κίτρινων και κόκκινων σφαιριδίων.



Εικόνα 12:Ο Boxman

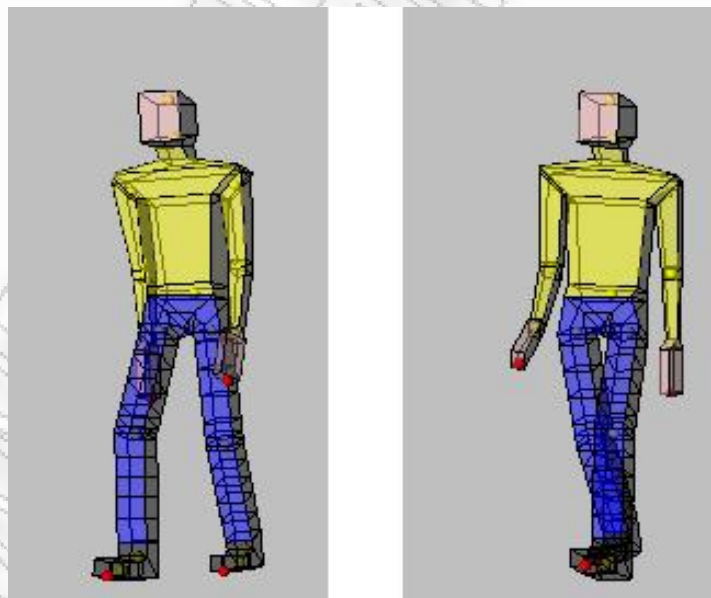
Για την πραγματοποίηση δοκιμών και τον έλεγχο της ακρίβειας των κινήσεων δημιουργήθηκε ένα interface για το X3d με κουμπιά για την απευθείας διαχείριση των animations.



Εικόνα 13: Ο χρήστης μπορεί να επιλέξει animation (περπάτημα ή κίνηση του χεριού)

6.1.1 Walk Animation

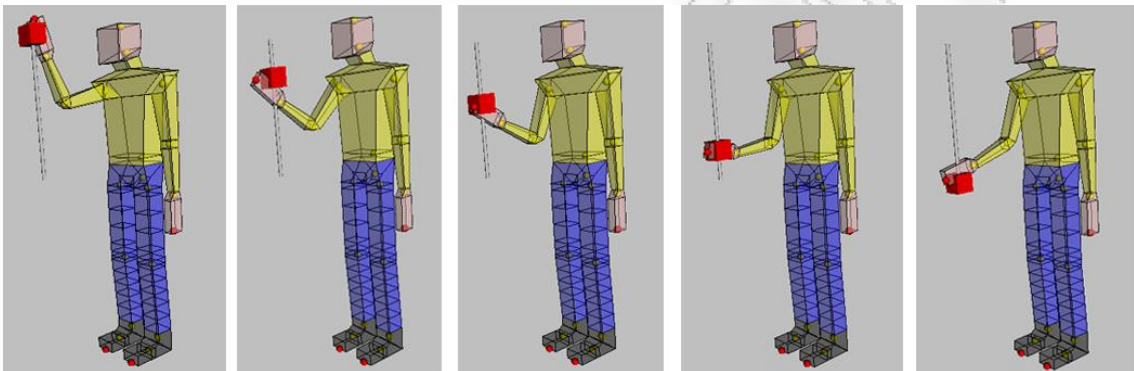
Το WalkAnimation αναπαριστά το ανθρώπινο περπάτημα. Ο χρήστης μπορεί να παρέμβει στην ταχύτητα του βαδίσματος και να την αυξήσει ή να τη μειώσει σύροντας το κόκκινο κουμπί αριστερά ή δεξιά αντίστοιχα.



Εικόνα 14: Walk Animation

6.1.2 Touch Animation

Το TouchAnimation δίνει τη δυνατότητα της κίνησης του χεριού του πράκτορα για να ακουμπήσει κάποιο αντικείμενο που βρίσκεται σε κοντινή απόσταση. Το εύρος της κίνησης που μπορεί να πραγματοποιήσει ο πράκτορας είναι περιορισμένο, όπως άλλωστε η κίνηση ενός πραγματικού ανθρώπινου χεριού. Για να ακουμπήσει ο χαρακτήρας κάποιο αντικείμενο πρέπει να βρίσκεται σε απόσταση 2 με αυτό και το αντικείμενο να έχει ύψος περίπου από τη μέση του μέχρι λίγο πάνω από το κεφάλι του.



Εικόνα 15:Κίνηση του χεριού σε διάφορες θέσεις

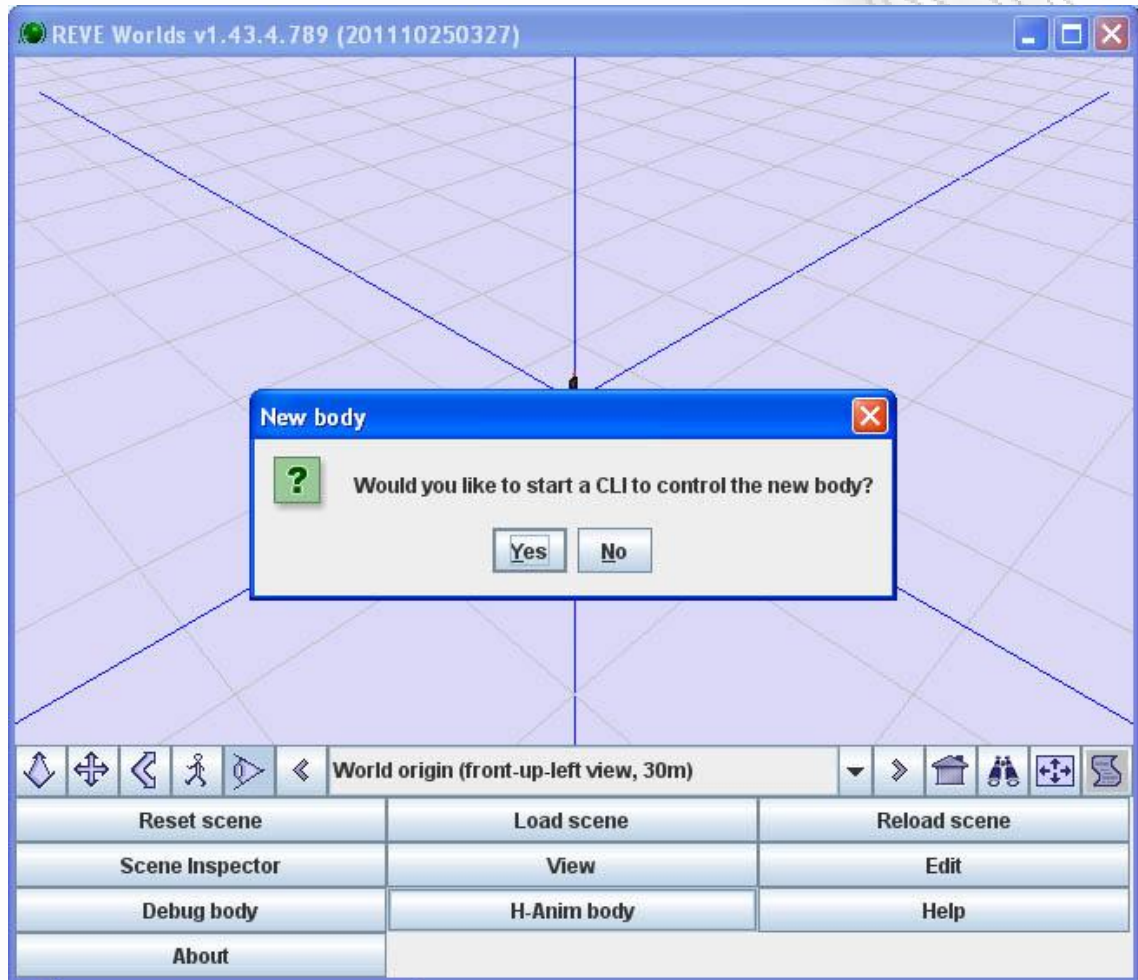
Η λειτουργία της κίνησης του χεριού ελέγχεται με το κόκκινο κουμπί που βρίσκεται κοντά στο χέρι του Boxman. Όταν το κόκκινο κουμπί μετακινείται, ο Boxman λυγίζει τον ώμο και τον αγκώνα του για να φέρει το χέρι του στο σημείο που βρίσκεται το κουμπί.

Κόμβος ελέγχου animation	Πεδίο εκκίνησης	Τιμή εκκίνησης	Πεδίο παύσης	Τιμή παύσης
WALK_ANIMATOR	enabled	true	enabled	false
TOUCH_ANIMATOR	enabled	true	enabled	false

Κόμβος ελέγχου animation	Λειτουργία	Πεδίο	Τιμή
WALK_ANIMATOR	Αλλαγή ταχύτητας	cycleInterval	SFTime
TOUCH_ANIMATOR	Κίνηση χεριού	POINTinput	SFVec3f

6.2 H-ANIM πράκτορες στο REVE Worlds

Το περιβάλλον διεπαφής του REVE Worlds έχει πλέον ένα νέο κουμπί που επιτρέπει στους χρήστες να φορτώνουν ένα σώμα H-ANIM. Όταν τελειώσει η διαδικασία του φορτώματος ο χρήστης διερωτάται αν θέλει να αρχίσει ένα CLI (Command-Line Interface) για τον έλεγχο του σώματος.



Εικόνα 16: Το περιβάλλον REVE Worlds με τη δυνατότητα φόρτωσης ενός σώματος H-ANIM.

6.2.1 Walk Animation

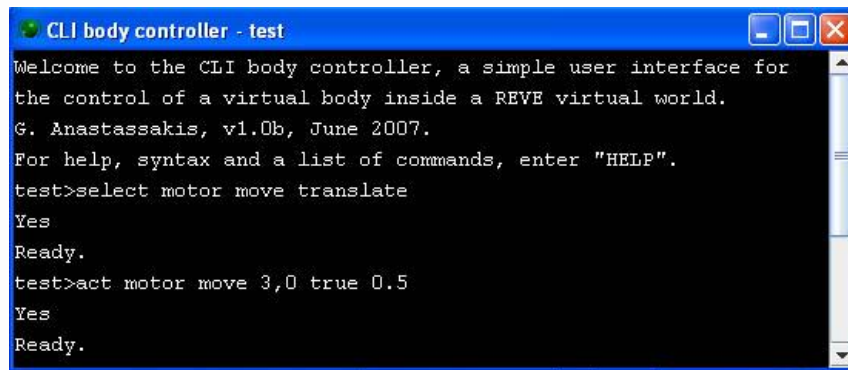
Μέσω του CLI ο χρήστης μπορεί να ελέγξει την κίνηση του μοντέλου. Συγκεκριμένα, με την εντολή

```
Select motor move translate
```

Επιλέγει την κίνηση για περπάτημα και γράφοντας

```
Act motor move 3,0 true 0.5
```

Δίνει την εντολή στον πράκτορα να κινηθεί 3 μέτρα με ταχύτητα 0,5 μέτρα/sec.

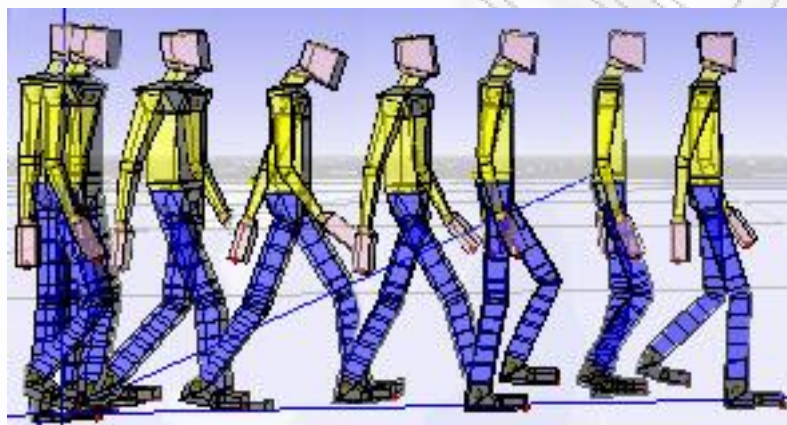


```

CLI body controller - test
Welcome to the CLI body controller, a simple user interface for
the control of a virtual body inside a REVE virtual world.
G. Anastassakis, v1.0b, June 2007.
For help, syntax and a list of commands, enter "HELP".
test>select motor move translate
Yes
Ready.
test>act motor move 3,0 true 0.5
Yes
Ready.

```

Εικόνα 17: Οι εντολές για περπάτημα στο CLI



Εικόνα 18: Το Walk Animation στο REVE Worlds

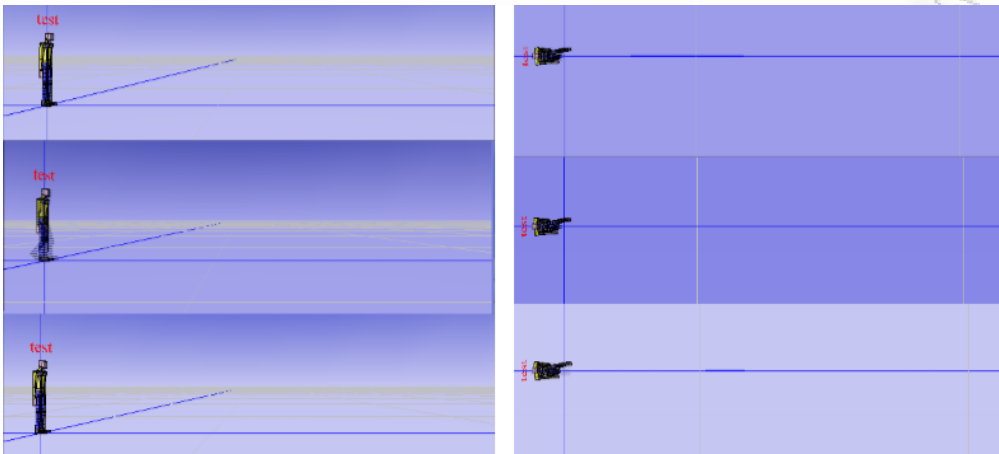
6.2.2 Σύγκριση του Walk Animation για διαφορετικές ταχύτητες

Για την επίδειξη του Walk Animation στο REVE Worlds έγινε μια σύγκριση μεταξύ των κινήσεων τριών πρακτόρων που έχουν διαφορετικές ταχύτητες.

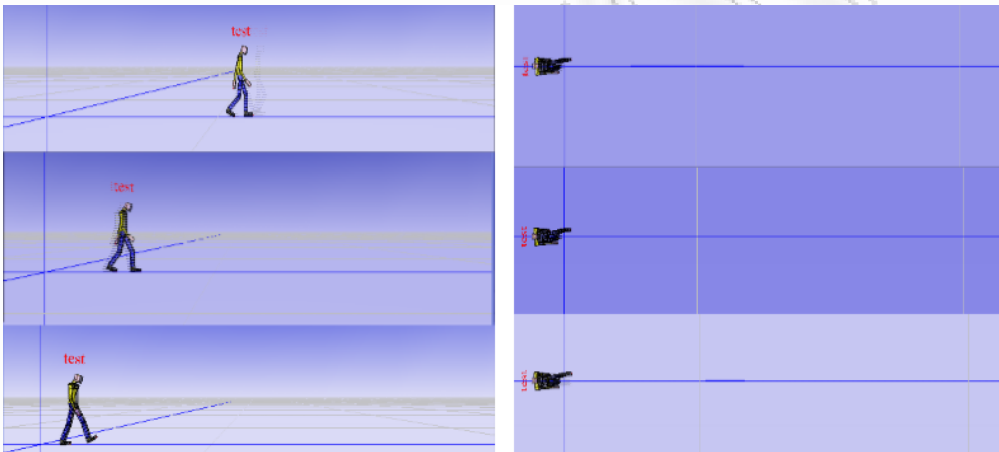
Ταχύτητα πράκτορα	Χρόνος τερματισμού
0.5	4 sec
1	8 sec
3	18 sec

Πίνακας 1: Οι ταχύτητες των τριών πρακτόρων και ο χρόνος τερματισμού τους

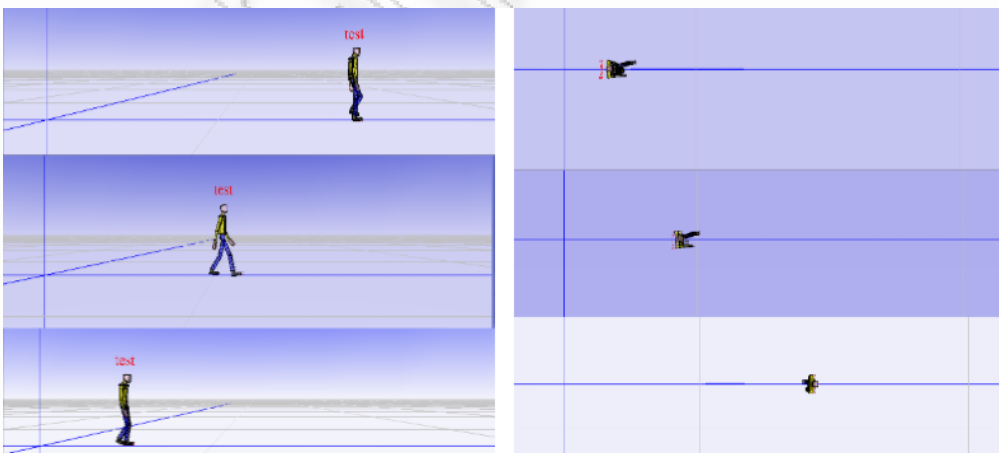
Παρακάτω παρουσιάζονται στιγμιότυπα τους σε διαφορετικά χρονικές στιγμές.



Εικόνα 19:Οι τρεις πράκτορες στην εκκίνησή τους στα 0 sec



Εικόνα 20:Οι τρεις πράκτορες στη χρονική στιγμή 1 sec



Εικόνα 21:Οι τρεις πράκτορες στη χρονική στιγμή 5 sec



Εικόνα 22:Οι τρεις πράκτορες στη χρονική στιγμή 8 sec



Εικόνα 23:Οι τρεις πράκτορες στη χρονική στιγμή 13 sec



Εικόνα 24:Στα 18 sec έχει τερματίσει και ο τελευταίος πράκτορας

ΚΕΦΑΛΑΙΟ 7

Συμπεράσματα

7.1 Συμπεράσματα

Η διατριβή αυτή αποτελεί το πρώτο βήμα για τη δημιουργία μιας ολοκληρωμένης βιβλιοθήκης με animations για τους πράκτορες του REVE Worlds. Αφού μελετήθηκαν κάποιοι τρόποι που οι εικονικοί πράκτορες μπορούν να αποκτήσουν κίνηση, αναπτύχθηκε ένα πλαίσιο που βασίζεται στο H-ANIM, ένα εύχρηστο, ανοικτό και δωρεάν πρότυπο που επιτρέπει την γρήγορη και εύκολη παραγωγή animations χωρίς ιδιαίτερες τεχνικές γνώσεις. Για την επίδειξη της ορθής λειτουργίας του πλαισίου παρουσιάστηκαν δύο παραδείγματα κίνησης (περπάτημα και αφή) τα οποία συνδέθηκαν με το REVE Worlds μέσω κλάσεων java. Αυτό επιτρέπει στα μοντέλα και τις κινήσεις να είναι ανεξάρτητα από το εικονικό περιβάλλον και να είναι εύκολη η επεξεργασία, η ενσωμάτωση ή η αφαίρεσή τους.

7.2 Μελλοντική εργασία

Στα παραδείγματα εκτέλεσης παρουσιάστηκαν ανθρωπόμορφα μοντέλα με σχετικά περιορισμένο εύρος κινήσεων λόγω του μικρού αριθμού αρθρώσεων που σχηματίζουν τη σκελετική δομή τους. Πιο λεπτομερής κίνηση μπορεί να δημιουργηθεί με την κατάλληλη επέκταση, που θα επιτρέπει τη χρήση μοντέλων με μεγαλύτερο LOA. Έτσι μπορούν, για παράδειγμα, να αποκτήσουν δυνατότητες κίνησης και τα δάχτυλα των χεριών.

Άλλος ένας στόχος είναι να ενσωματωθούν περισσότερα animations στον κόσμο του Reve ώστε η κίνηση των πρακτόρων να είναι όσο γίνεται ρεαλιστική και πλήρης. Η δημιουργία λογισμικού που παράγει αυτόματα τις java κλάσεις από τα αρχεία X3d θα μπορούσε να επισπεύσει σημαντικά αυτή τη διαδικασία. Οι πράκτορες δεν είναι απαραίτητο να περιορίζονται στην ανθρώπινη μορφή, αλλά θα μπορούν να παίρνουν κι άλλες μορφές, όπως ζώα.

8 ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ

- [1] D. Thalmann. "Challenges for the Research in Virtual Humans", In *Proceedings Workshop on Achieving Human-Like Behaviour in Interactive Animated Agents*, AGENTS 2000, Barcelona, Spain
- [2] P.Kalra , N. Magnenat-Thalmann , L. Moccozet , G. Sannier , A. Aubel and D. Thalmann, "Real-Time Animation of Realistic Virtual Humans", *IEEE Computer Graphics and Applications*, v.18 n.5, p.42-56, September 1998
- [3] CAESAR anthropometric data, <http://store.sae.org/caesar/>
- [4] A. Albin-Clark and T.L.J. Howard, Automatically generating virtual humans using evolutionary algorithms, In *Proceedings of Theory and Practice of Computer Graphics*, Cardiff, 2009. Eurographics.
- [5] Humanoid Animation Specification <http://www.web3d.org/x3d/specifications/ISO-IEC-19774-HumanoidAnimation/>
- [6] M.Preda and F.Preteux, "Virtual Character Within MPEG-4 Animation Framework eXtension", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 7, July 2004
- [7] D. Thalmann and F. Vexo, MPEG-4 Character Animation, Virtual Reality Laboratory, Swiss Federal Institute of technology, . 2003.
- [8] D.Nikovski, "Dynamic simulation methods for animation of legged locomotion". *CS 585 term paper*, 1994, SIU
- [9] M.Gillies and B.Spanlang, "Comparing and Evaluating Real Time Character Engines for Virtual Environments", *Presence: Teleoperators and Virtual Environments*, ,Vol. 19, No. 2, Pages 95-117, April 2010
- [10] J.Lee, J.Chai, P. S. A. Reitsma, J. K. Hodgins and N. S. Pollard, Interactive control of avatars animated with human motion data. In *Proceedings of ACM SIGGRAPH 2002*, Annual Conference Series, ACM SIGGRAPH.
- [11] L. Kovar , M. Gleicher and F. Pighin, Motion graphs, *ACM SIGGRAPH 2008 classes*, August 11-15, 2008, Los Angeles, California
- [12] J. Gästrin, Physically Based Character Simulation – Rag Doll Behaviour in Computer Games, Royal Institute of Technology, 2004, Stockholm
- [13] P.Lih-Hern, T.Y. Siang, W.C. Foo, and W.L. Kuan, ODECAL, a Flexible Open Source Rag Doll Simulation Engine, *Lecture Notes in Computer Science*, 3942, 680-687, 2006
- [14] S. Kshirsagar , N. Magnenat-Thalmann , A. Guye-Vuillème , D. Thalmann , K. Kamyab and E. Mamdani, Avatar Markup Language, *Proceedings of the workshop on Virtual environments 2002*, May 30-31, 2002, Barcelona, Spain

- [15] Y. Arafa and A. Mamdani, Scripting embodied agents behaviour with CML: character markup language, *Proceedings of the 8th international conference on Intelligent user interfaces*, January 12-15, 2003, Miami, Florida, USA
- [16] N. I. Badler , R. Bindiganavale , J. Allbeck , W. Schuler , L. Zhao and M. Palmer, Parameterized action representation for virtual human agents, *Embodied conversational agents*, MIT Press, Cambridge, MA, 2001
- [17] Z. Huang, A. Elien and C.Visser, STEP: A Scripting Language for Embodied Agents, *Proceedings of the Workshop on Lifelike Animated Agents*, 2002, Piez
- [18] Z. Huang , A. Eliëns and C. Visser, XSTEP: A Markup Language for Embodied Agents, *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA 2003)*, p.105, May 08-09, 2003
- [19] S Kopp, B. Krenn, S. Marsella, A. Marshall, C. Pelachaud, H. Pirker, K. Thórissonand and H. Vilhjalmsson, Towards a Common Framework for Multimodal Generation in ECAs: The Behavior Markup Language, In *Proceedings of the 6th International Conference on Intelligent Virtual Agents*, Marina del Rey, CA, August 21-23, 2005-21, 2006
- [20] VHML www.vhml.org
- [21] H. Vilhjálmsson , N. Cantelmo , J. Cassell , N. E. Chafai , M. Kipp , S. Kopp , M. Mancini , S. Marsella , A. N. Marshall , C. Pelachaud , Z. Ruttkay , K. R. Thórisson , H. Welbergen and R. J. Werf, The Behavior Markup Language: Recent Developments and Challenges, *Proceedings of the 7th international conference on Intelligent Virtual Agents*, September 17-19, 2007, Paris, France
- [22] Y. Arafa , K. Kamyab and E. Mamdani, "Character animation scripting languages: a comparison", *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, July 14-18, 2003, Melbourne, Australia
- [23] M. Gillies, and B. Spanlang, "Comparing and evaluating realtime character engines for virtual environments," *PRESENCE -Teleoperators and Virtual Environments*, vol. in press, 2009.
- [24] Ieronutti, L., & Chittaro, L., Employing virtual humans for education and training in X3D/VRML worlds. *Computers & Education*, 49(1), 93–109, 2007
- [25] F. Cordier, H. Seo, N. Magnenat Thalmann. Made-to measure technologies for an online clothing store. *IEEE Computer Graphics and Applications*;23(1):38–48, 2003
- [26] Ma, Minhua and Paul Mc Kevitt. 2003. Building character animation for intelligent storytelling with the H-Anim standard. *Eurographics Ireland Chapter Workshop Proceedings* , M. McNeill (Ed.), 9-15, Coleraine, 2003

- [27] M. Evers and A. Nijholt, Jacob - an animated instruction agent for virtual reality. In *Advances in Multimodal Interfaces - ICMI 2000, Proc. Third International Conference on Multimodal Interfaces* 526–533, 2000
- [28] K. Karpouzis, G. Caridakis, S.E. Fotinea, and E. Efthimiou, Educational resources and implementation of a Greek sign language synthesis architecture, *Computers & Education*, 49(1), 54–74, 2007
- [29] H-ANIM workshop <http://hclab.uniud.it/let-web3d/H-Anim-2004.pdf>
- [30] X3D Specification <http://web3d.org/x3d/specifications/ISO-IEC-19775-1.2-X3D-AbstractSpecification/>
- [31] X3D Overview <http://www.web3d.org/about/overview/>
- [32] Kambi VRML game engine <http://vrmlengine.sourceforge.net/index.php>
- [33] REVE <http://kelnet.cs.unipi.gr/reve/>
- [34] REVE presentation
http://www.iit.demokritos.gr/index.php?option=com_content&view=article&id=169&Itemid=80
- [35] REVE presentation <http://www.iit.demokritos.gr/docs/seminars/anastassakis-IIT-20101013.pdf>
- [36] http://web.itu.edu.tr/~pazarci/mpeg4/MPEG_Overview1_w2196.htm
- [37] <http://coven.lanacs.ac.uk/mpeg4/>
- [38] <http://hclab.uniud.it/h-animator/images/Keyframer.jpg>
- [39] http://h-anim.org/h-anim1_1f.gif
- [40] <http://www.inrialpes.fr/sed/people/boissieux/HUMANS/H-ANIM/>
- [41] <http://www.iit.demokritos.gr/docs/seminars/anastassakis-IIT-20101013.pdf>
- [42] <http://kelnet.cs.unipi.gr/reve/files/images/R2D2InHouse.JPG>
- [43] <http://www.web3d.org/x3d/content/examples/HumanoidAnimation/BoxMan.x3d>
- [44] http://www.web3d.org/x3d/content/examples/HumanoidAnimation/_pages/page18.html

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΡΑΙΑ