



## Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών  
«Προηγμένα Συστήματα Πληροφορικής»

### Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	<b>Εφαρμογή Reve World Designer - Επιλογή τρισδιάστατου αντικειμένου για τη σύνθεση εικονικών κόσμων Reve World Designer Application - Selection of a 3D object for composing a virtual world</b>
Όνοματεπώνυμο Φοιτητή	<b>Γεώργιος Σανιόγλου</b>
Πατρώνυμο	<b>Μενέλαος</b>
Αριθμός Μητρώου	<b>ΜΠΣΠ/12066</b>
Επιβλέπων	<b>Θεμιστοκλής Παναγιωτόπουλος, Καθηγητής</b>

### **Τριμελής Εξεταστική Επιτροπή**

Θεμιστοκλής  
Παναγιωτόπουλος  
Καθηγητής

Γεώργιος Τσιχριντζής  
Καθηγητής

Δημήτριος Αποστόλου  
Επίκουρος Καθηγητής

## **Ευχαριστίες**

Η παρούσα διπλωματική διατριβή πραγματοποιήθηκε στα πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών, στα Προηγμένα Συστήματα Πληροφορικής στην Πληροφορική του Πανεπιστημίου Πειραιώς. Πρωτίστως, θα ήθελα να ευχαριστήσω θερμά τους επιβλέποντες καθηγητές, κύριο Θεμιστοκλή Παναγιωτόπουλο και κύριο Γεώργιο Αναστασάκη, για την πολύτιμη καθοδήγησή τους, την προτροπή, τη συνεργασία και το αληθινό τους ενδιαφέρον. Επίσης, ευχαριστώ θερμά όλους τους ανθρώπους που στη διάρκεια των σπουδών μου συνέβαλλαν με τέτοιο τρόπο ώστε να με πείσουν πως η μόρφωση είναι μία διαρκής και αέναη διαδικασία. Οφείλω θερμές ευχαριστίες ιδιαίτερα σε πρώτο βαθμό στην μητέρα μου και στον αδερφό μου, αλλά και σε όλη την οικογένεια μου για τη στήριξή τους, καθώς και σε όσους από το φιλικό και συναδελφικό περιβάλλον με πραγματικό ενδιαφέρον στήριξαν την προσπάθειά μου. Τέλος ευχαριστώ θερμά την συνάδερφο Έννα Τσελεμέγκου για την καταπληκτική συνεργασία και την απεριόριστη υπομονή καθ' όλη την διάρκεια εκπόνησης και σύνδεσης των διπλωματικών διατριβών μας.

## Περίληψη

Αντικείμενο της παρούσας διπλωματικής διατριβής ήταν η συμβολή στην ανάπτυξη ενός γραφικού περιβάλλοντος, μιας εφαρμογής, με στόχο την επεξεργασία και επιλογή κατάλληλων τρισδιάστατων αντικειμένων για την σύνθεση εικονικών κόσμων για την πλατφόρμα REVE. Με την επιλογή των κατάλληλων τρισδιάστατων αντικειμένων είναι εφικτή σε πρώτο στάδιο η δημιουργία ευφυών εικονικών περιβαλλόντων. Τα ευφυή εικονικά περιβάλλοντα προϋποθέτουν την σχεδίαση εικονικών κόσμων, οπότε η εν λόγω εργασία στοχεύει στην διευκόλυνση του χρήστη κατά την επιλογή των σωστών αντικειμένων, ολόκληρου του αντικείμενου ή μέρος του αντικείμενου. Σκοπός της συγκεκριμένης διπλωματικής διατριβής ήταν να δίνεται η δυνατότητα στον χρήστη να βλέπει, να επιλέγει και να επεξεργάζεται άμεσα τρισδιάστατα αντικείμενα που έχουν ως στόχο την σύνθεση ενός εικονικού κόσμου. Σαν εικονικός κόσμος ορίζεται ένα σύνολο από εικονικά αντικείμενα. Τα αντικείμενα αυτά αναπαριστώνται με τη βοήθεια των ηλεκτρονικών υπολογιστών και μπορεί να είναι φυσικά ή φανταστικά πρόσωπα ή/και πράγματα και διακρίνονται από κάποια βασικά χαρακτηριστικά ή να έχουν κάποιες χαρακτηριστικές ιδιότητες που να τα καθιστούν μοναδικά στον χρήστη, όπως οι διαστάσεις τους, το σχήμα τους, το χρώμα τους και η θέση τους. Για την σχεδίαση των εικονικών κόσμων χρησιμοποιείται η αναπαράσταση REVE που ορίζει έναν κόσμο ως σύνολο από items. Η διατριβή εστιάζει σε τρισδιάστατα αντικείμενα τύπου VRML και X3D. Η REVE χρησιμοποιεί την γλώσσα VERL για την αναπαράσταση του κόσμου και ως εκ τούτου η εφαρμογή δίνει τη δυνατότητα προεπεξεργασίας τέτοιων αντικειμένων με στόχο την σωστή και γρήγορη επιλογή των κατάλληλων αντικειμένων. Η τεχνολογία που χρησιμοποιήθηκε είναι η Java SE 7 και JDK 7, με εκτενή χρήση των βιβλιοθηκών swing, jaxp και xj3d, για την ανάπτυξη του γραφικού περιβάλλοντος, τη δημιουργία και επεξεργασία των veril αρχείων και την απεικόνιση των αντικειμένων και του κόσμου αντίστοιχα.

Λέξεις κλειδιά: REVE, item, εικονικός κόσμος, εικονικό περιβάλλον, τρισδιάστατα αντικείμενα, XJ3D, veril, σχεδίαση εικονικών κόσμων

## Abstract

The purpose of this MSc thesis was to contribute to the development of a graphical user interface, an application, aimed the processing and selection of the suitable three-dimensional objects for the synthesis of virtual worlds for the platform of REVE. By choosing the appropriate three-dimensional objects it is possible in the first step to create intelligent virtual environments. Intelligent Virtual Environments require the design of virtual worlds, thus, this research aims in assisting the user in that aspect, in selecting the correct objects, whole the object or part of the object. The purpose of this MSc thesis was to enable the user to directly view, select and edit three-dimensional objects for the construction of a virtual world. As virtual world is defined a set of virtual objects. The objects are represented with the aid of computer and may be real or imaginary beings and/or things and have some characteristic properties that make them unique to the user, such as their size, shape, their color and position. For the design of virtual worlds, REVE use a world as a set of items. This MSc thesis focuses on three-dimensional VRML and X3D type objects. REVE uses the VERL language for representing the virtual worlds and thus the application gives you the opportunity preprocessing such items with the aim of correct and quickly select the appropriate objects. The technology used is the Java SE 7 and SDK 7, with an extent use of the swing library for developing the graphic environment, of the jaxp library for processing the veril files and the xj3d library for visualizing the items and the world.

Keywords: REVE, item, virtual world, virtual environment, three-dimensional objects, XJ3D, veril, virtual world designer

**Περιεχόμενα**

Ευχαριστίες.....	3
Περίληψη.....	4
Abstract.....	4
Εισαγωγή .....	8
Κεφάλαιο 1 - Εισαγωγή Στην Εικονική Πραγματικότητα.....	11
1.1 Τι Είναι Εικονική Πραγματικότητα.....	11
1.2 Η Εικονική Πραγματικότητα Στα Γραφικά 3D .....	12
1.3 Τεχνολογίες Αναπαράστασης.....	12
1.4 Συστατικά Συστήματος Εικονικής Πραγματικότητας.....	19
1.5 Η Εικονική Πραγματικότητα Ως Πρακτική Εφαρμογή.....	20
1.6 Το Μέλλον Της Εικονικής Πραγματικότητας .....	21
Κεφάλαιο 2 - Εικονικά Περιβάλλοντα .....	22
2.1 Ανάπτυξη Και Κατηγορίες Εικονικών Περιβαλλόντων.....	22
2.2 Λειτουργικά Χαρακτηριστικά Εικονικού Περιβάλλοντος.....	22
2.3 Εφαρμογές Εικονικών Περιβαλλόντων.....	23
2.4 Ευφύες Εικονικό Περιβάλλον .....	26
2.5 Ελλείψεις Στον Τομέα Των Εικονικών Περιβαλλόντων.....	26
Κεφάλαιο 3 - Αλληλεπίδραση Χρήστη - Εικονικού Περιβάλλοντος .....	28
3.1 Διακρίσεις Εικονικού Περιβάλλοντος.....	28
3.2 Συστήματα Γραφικών.....	29
3.3 Λίγα Λόγια Για Τα Γραφικά Η/Υ .....	29
3.4 Η Λειτουργία Buffering .....	33
<b>3.5 Η Διαδικασία Του Rendering .....</b>	<b>33</b>
3.6 Γράφημα Σκηής.....	33
Κεφάλαιο 4 - Εικονικοί - Ευφύης Πράκτορες .....	37

4.1	Τι Είναι Πράκτορας .....	37
4.2	Εικονικοί Πράκτορες Σήμερα.....	37
4.3	Διαχωρισμός Πρακτόρων .....	38
4.3.1	Φυσικοί Πράκτορες.....	38
4.3.2	Γνωσιακοί Πράκτορες .....	39
4.3.3	Ευφυής Πράκτορας.....	39
4.4	Άλλες Κατηγορίες Πρακτόρων .....	40
Κεφάλαιο 5 - Εικονικός Κόσμος.....		41
5.1	Ο Εικονικός Κόσμος Second Life.....	42
5.2	Εφαρμογές Των Εικονικών Κόσμων .....	44
Κεφάλαιο 6 - Γλώσσες Αναπαράστασης Εικονικών Κόσμων .....		46
6.1	Η Γλώσσα Τρισδιάστατων Γραφικών OpenGL .....	46
6.2	Γλώσσα Προγραμματισμού VRML .....	47
6.3	Αρχιτεκτονική Αναπαράστασης Τρισδιάστατων Γραφικών X3D.....	50
6.4	Γλώσσα Προγραμματισμού JAVA 3D.....	52
ΚΕΦΑΛΑΙΟ 7 - Εργαλεία Σχεδίασης Εικονικών Κόσμων .....		53
7.1	Η Εφαρμογή Unity .....	53
7.2	Η Εφαρμογή 3DStudioMax.....	56
Κεφάλαιο 8 - Η Πλατφόρμα Reve .....		67
8.1	Τι Είναι Η Πλατφόρμα REVE .....	67
Κεφάλαιο 9 - Η Γλώσσα VERL .....		69
9.1	Τι Είναι Η Γλώσσα VERL.....	69
Κεφάλαιο 10 - Η Πλατφόρμα REVEWORLDS .....		78
Κεφάλαιο 11 - Σχεδίαση Της Εφαρμογής – Ανάλυση Απαιτήσεων.....		83
11.1	Ανάλυση Απαιτήσεων .....	83
11.2	Τεχνολογίες Υλοποίησης .....	85

Κεφάλαιο 12 - Υλοποίηση Εφαρμογής Για Την Επιλογή Τρισδιάστατων Αντικειμένων Για Τη Σύνθεση Εικονικών Κόσμων .....	92
12.1 Δομή Εφαρμογής .....	92
12.2 Η γραμμή Εργαλείων (Κεντρικό Μενού).....	94
12.2.1 Εύρεση Και Άνοιγμα Τρισδιάστατων Αντικειμένων Με Την Επιλογή «OpenFile».....	95
12.2.2 Εισαγωγή Ενός Αντικειμένου Με Την Επιλογή «AddNodetoWorld».....	96
12.2.3 Επιλογή «Πληροφορίες Εφαρμογής» .....	97
12.2.4 Επιλογή «Βοήθεια».....	98
12.2.5 Έξοδος Από Την Εφαρμογή .....	99
12.3 Κατάλογος Αρχείων Του Χρήστη .....	100
12.4 Παράθυρο Απεικόνισης Τρισδιάστατων Αντικειμένων.....	101
12.5 Παράθυρο Γραφήματος Σκηνής .....	102
12.6 Ορισμός Του Item .....	106
Κεφάλαιο 13 - Συμπεράσματα.....	110
Βιβλιογραφία .....	114

## Εισαγωγή

Στην παρούσα διπλωματική διατριβή στόχο έχουμε να ασχοληθούμε με την ανάπτυξη μιας εφαρμογής που συνθέτει τρισδιάστατα αντικείμενα. Τελικός στόχος ήταν η ανάπτυξη ενός γραφικού περιβάλλοντος επιλογής τρισδιάστατων αντικειμένου για τη σύνθεση εικονικών κόσμων για την πλατφόρμα REVE. Η πλατφόρμα REVE (Representation and Execution of Virtual Environments) έχει αναπτυχθεί από το Εργαστήριο Τεχνολογίας Γνώσης του Πανεπιστημίου Πειραιώς και αποτελείται από ένα σύνολο εφαρμογών και προγραμματιστικών εργαλείων, τα οποία επιτρέπουν την ανάπτυξη εφαρμογών ευφυών εικονικών περιβαλλόντων (intelligent virtual environments, IVEs). Η αλληλεπίδραση γίνεται μέσω ευφυών πρακτόρων στο περιβάλλον της reve. Αρχικά θα αναλύσουμε και θα εξοικειωθούμε με τις βασικές έννοιες με τις οποίες καταπιανόμαστε, θα αναλυθούν ορισμοί που θα βοηθήσουν στην κατανόηση και στην διάκριση των αντικειμένων που αναφέρονται στην παρούσα διπλωματική εργασία. Έπειτα θα αναλυθεί ο τρόπος σχεδίασης της εφαρμογής, η λειτουργικότητα της και τέλος θα παρουσιαστούν όλες οι δυνατότητες της.

Ένα εικονικό περιβάλλον αποτελείται από περιεχόμενο (αντικείμενα και δράστες (actors)), γεωμετρία και δυναμική, με ένα εγωκεντρικό πλαίσιο αναφοράς, που περιλαμβάνει την αντίληψη των αντικειμένων σε βάθος και που εγείρει διάφορες αισθήσεις ταυτόχρονα καθώς και την αλληλεπίδραση ενός ή παραπάνω χρηστών με αυτό. Το ευφύες εικονικό περιβάλλον προσθέτει δυναμικότητα στο εικονικό περιβάλλον, εισάγοντας το στοιχείο της τεχνητής νοημοσύνης. Όπως είναι λογικό σε ένα ευφύες εικονικό περιβάλλον συναντάμε οντότητες που αναπαρίστανται και δρουν αυτόνομα. Στην περίπτωση της πλατφόρμας REVE, αυτές οι οντότητες αναπαρίστανται με την μορφή ενσώματων ευφυών πρακτόρων που μέσω αισθητήρων (sensors) λαμβάνουν πληροφορία για τον κόσμο, ενώ μέσω επιδραστών (effectors) αλληλεπιδρούν με αυτόν, μεταβάλλοντάς τον.

Βασικό συστατικό των εικονικών περιβαλλόντων, ευφυών και μη, είναι ο εικονικός κόσμος. Ο εικονικός κόσμος αποτελείται από ένα σύνολο εικονικών αντικειμένων και βασικό στοιχείο ενός εικονικού κόσμου είναι η «πιστότητα» (believability), που επηρεάζει και τον βαθμό εμπύθισης ενός χρήστη σε αυτόν. Τα εικονικά αντικείμενα μπορεί να έχουν οποιαδήποτε μορφή ανάλογα το περιεχόμενο του κόσμου και διακρίνονται από κάποια βασικά χαρακτηριστικά που τα καθιστούν μοναδικά και αναγνωρίσιμα από τον χρήστη. Αυτά είναι η γεωμετρία τους, οι διαστάσεις τους, το μέγεθός τους, το χρώμα τους, το υλικό τους, η υφή τους και η θέση τους στο χώρο. Πέραν των φυσικών χαρακτηριστικών που οπτικοποιούν τα εικονικά αντικείμενα, αυτά μπορούν να διακριθούν και από άλλα χαρακτηριστικά που διεγείρουν τις υπόλοιπες αισθήσεις του χρήστη, όπως την ακοή ή την αφή. Ωστόσο, μια τέτοια αναπαράσταση των αντικειμένων προϋποθέτει πιο εξελιγμένη τεχνολογία. Η «πιστότητα» συνεπάγεται πως ο εικονικός κόσμος πρέπει να διέπεται από ένα σύνολο νόμων και κανόνων, που είτε προσομοιώνουν τους φυσικούς είτε ορίζουν δικούς τους νόμους. Η «πιστότητα» σε έναν εικονικό κόσμο είναι σημαντική, γιατί επηρεάζει την αντίληψη του χρήστη κατά την αλληλεπίδρασή του με τον κόσμο, ενώ ορίζει και τον τρόπο συμπεριφοράς των ευφυών οντοτήτων σε αυτόν.

Ως εικονικό κόσμο μπορούμε να ορίσουμε την αναπαράσταση ενός τεχνητού κόσμου που προκύπτει ως αποτέλεσμα της επίδρασης των εικονικών περιβαλλόντων και ο οποίος αποτελείται από ένα σύνολο εικονικών αντικειμένων, προκαλώντας στον χρήστη την εντύπωση πως βρίσκεται μέσα στον κόσμο, μέσα από τη διέγερση μέρους ή όλων των αισθήσεών του. Η δημιουργία, λοιπόν, του εικονικού κόσμου είναι, ουσιαστικά, το πρώτο βήμα σε κάθε προσπάθεια ανάπτυξης μιας εφαρμογής ευφυούς εικονικού περιβάλλοντος.

Για την σχεδίαση των εικονικών κόσμων χρησιμοποιείται η αναπαράσταση REVE. Η πρωτοπορία στην REVE έγκειται, μεταξύ άλλων, στο ότι η αναπαράστασή του κόσμου αποδεδμεύεται από τις λεπτομέρειες της υλοποίησής του και ο κόσμος ορίζεται σε ένα εννοιολογικό πλαίσιο που περιλαμβάνει υψηλού επιπέδου έννοιες, όπως το γράφημα σκηνής, το αντικείμενο, η φυσική υπόσταση του αντικειμένου, η σημασιολογία του και η ικανότητα αλληλεπίδρασης ενός χρήστη με αυτό.

Η γεφύρωση μεταξύ εννοιολογικής προσέγγισης και υλοποίησης του κόσμου επιτυγχάνεται μέσα από την βασισμένη σε XML γλώσσα VERL (Virtual Environment Representation Language). Η verl ορίζει τον εικονικό κόσμο μέσα από ένα σύνολο από <items>



που αποτελούν τον κόσμο, δίνοντας στοιχεία για την προέλευσή τους και τα χαρακτηριστικά τους, εκμεταλλευόμενα την αναπαράσταση x3d και vml αντικειμένων μέσω γραφήματος σκηνής για την επιλογή των items και την δομή της xml για την αναπαράσταση του κόσμου. Τα δεδομένα σε γλώσσα xml φορτώνονται από το σύστημα REVE Worlds, το οποίο οπτικοποιεί τον εικονικό κόσμο σε τρεις διαστάσεις, επιτρέπει την περιήγηση του χρήστη με διάφορους τρόπους, την σύνδεση εξωτερικών εφαρμογών παραγωγής συμπεριφοράς με εικονικά σώματα με στόχο την εισαγωγή ευφυών εικονικών πρακτόρων (intelligent virtual agents, IVAs) στον εικονικό κόσμο.

Στην παρούσα διπλωματική διατριβή αναπτύσσουμε μια παραθυρική εφαρμογή που έχει ως στόχο να βοηθάει τον χρήστη στην κατάλληλη επιλογή των σωστών αντικειμένων και στην επεξεργασία τους με στόχο την σύνθεση ενός εικονικού κόσμου. Σε συνεργασία με την συνάδελφο Έντα Τσελεμέγκου και «παντρεύοντας» τις δύο διπλωματικές διατριβές έχουμε σαν τελικό αποτέλεσμα την μετατροπή αυτού του αντικειμένου σε item και την σχεδίαση του εικονικού κόσμου βάσει της αναπαράστασης REVE.

Στο πρώτο κεφάλαιο εισάγουμε τον χρήστη στις βασικές έννοιες της εικονικής πραγματικότητας. Εξηγούμε τι είναι εικονική πραγματικότητα, διακρίνουμε συσκευές που βοηθάνε τον χρήστη να συνδεθεί σε ένα εικονικό περιβάλλον και αναλύουμε τα συστατικά συστήματος της εικονικής πραγματικότητας.

Στο δεύτερο κεφάλαιο, αναφερόμαστε στα εικονικά περιβάλλοντα, αναλύουμε τα λειτουργικά χαρακτηριστικά ενός εικονικού περιβάλλοντος και κάνουμε μια αναφορά στις κατηγορίες των ευφυών εικονικών περιβαλλόντων, εφόσον τέτοια αναπτύσσει και η πλατφόρμα REVE, και τέλος αναφέρουμε κάποιες εφαρμογές των εικονικών περιβαλλόντων.

Στο τρίτο κεφάλαιο αναφερόμαστε στην αλληλεπίδραση του χρήστη με το εικονικό περιβάλλον, στις διακρίσεις που υπάρχουν στα εικονικά περιβάλλοντα και στα συστήματα γραφικών. Τέλος εξετάζουμε τους διαφορετικούς τρόπους αναπαράστασης εικονικών κόσμων, εστιάζοντας στο γράφημα σκηνής αφού τα αντικείμενα που χρησιμοποιούμε στην εφαρμογή μας για την σύνθεση του κόσμου έχουν σχεδιαστεί για να έχουν την μορφή γραφήματος σκηνής.

Στο τέταρτο κεφάλαιο, γίνεται μια εκτενή αναφορά στο τι είναι πράκτορες αλλά και μια αναφορά στα είδη των πρακτόρων που μπορούμε να συναντήσουμε στα εικονικά περιβάλλοντα.

Στο πέμπτο κεφάλαιο αρχικά αναφερόμαστε και αναλύουμε την έννοια του εικονικού κόσμου και στην συνέχεια παρουσιάζουμε ένα πολυχρηστικό περιβάλλον το οποίο αποτελεί και αιχμή στην εξέλιξη των εικονικών κόσμων.

Στο έκτο κεφάλαιο παρουσιάζουμε και αναλύουμε τις γλώσσες που χρησιμοποιούνται για την σχεδίαση των τρισδιάστατων γραφικών, OpenGL, VRML και X3D, αφού τα αντικείμενα που χρησιμοποιούμε στην εφαρμογή μας για την σύνθεση του κόσμου έχουν σχεδιαστεί βάσει αυτών και έχουν την μορφή γραφήματος σκηνής.

Στο έβδομο κεφάλαιο παρουσιάζουμε δυο πολυχρησιμοποιημένα και γνωστά εργαλεία σχεδίασης εικονικών κόσμων, τόσο για να εξετάσουμε υπάρχουσες υλοποιήσεις αντίστοιχων εργαλείων σχεδίασης όσο και για να λάβουμε υπόψη απαραίτητες λειτουργίες στην ανάλυση απαιτήσεων.

Στο όγδοο κεφάλαιο γίνεται μια εκτενή εισαγωγή στην πλατφόρμα REVE.

Στο ένατο κεφάλαιο επικεντρωνόμαστε στην γλώσσα xml, που θα αποτελέσει και την βάση σχεδίασης του κόσμου μας, καθώς και στο σύστημα Reve Worlds, στο οποίο είναι σκοπός να αναπαρίσταται ο τρισδιάστατος εικονικός κόσμος για την δημιουργία ευφυών εικονικών περιβαλλόντων.

Στο δέκατο κεφάλαιο θα γίνει μια αναφορά και μια εκτενή ανάλυση στην πλατφόρμα REVE Worlds. Η πλατφόρμα REVE Worlds είναι μια εφαρμογή που ενσαρκώνει την αναπαράσταση εικονικών κόσμων με βάση την αναπαράσταση REVE και η οποία επιτρέπει μια υψηλού επιπέδου αναπαράσταση του κόσμου

Στο ενδέκατο κεφάλαιο προχωράμε στη σχεδίαση της εφαρμογής, όπου, αρχικά, κάνουμε μια εκτίμηση για το τι θα θέλαμε να επιτύχουμε κατά την υλοποίηση και στη συνέχεια, αναλύουμε τις τεχνολογίες που αξιοποιήσαμε τελικά για να αναπτύξουμε την εφαρμογή.

Στο δωδέκατο κεφάλαιο κάνουμε μια αναλυτική περιγραφή της υλοποίησης της εφαρμογής, επικεντρωνόμενοι στις δυσκολίες που αντιμετωπίσαμε στην προσπάθεια αυτή. Η αναφορά των δυσκολιών έγινε τόσο για να αναδειχθεί το μέγεθος του χρόνου που απαιτούσε η τρέχουσα υλοποίηση, εμποδίζοντάς μας να εμπλουτίσουμε την εφαρμογή με επιθυμητές αλλαγές λόγω των περιορισμένων χρονικών πλαισίων, όσο και για να ληφθούν υπόψη για μελλοντικές αλλαγές και βελτιστοποίηση, ώστε να αποφευχθούν παρόμοια λάθη ή λανθασμένες προσεγγίσεις.

Τέλος, στο δέκατο τρίτο κεφάλαιο κάνουμε μια μικρή αναδρομή σε αυτά που επιτύχαμε σε σχέση με την ανάλυση απαιτήσεων και το πραγματικό αποτέλεσμα, καταθέτοντας τα συμπεράσματά μας και κάποιες ιδέες για ενδεχόμενες βελτιώσεις που μπορούν να πραγματοποιηθούν στο μέλλον.

Είναι σημαντικό να επισημάνουμε πως σκοπός της συγκεκριμένης διπλωματικής διατριβής, και πιο συγκεκριμένα ο συνδυασμός των δύο διπλωματικών διατριβών «Εφαρμογή Reve World Designer - Επιλογή τρισδιάστατου αντικειμένου για τη σύνθεση εικονικών κόσμων» και «Εφαρμογή Reve World Designer - Σχεδίαση εικονικών κόσμων για την πλατφόρμα REVE», δεν είναι η παροχή όλων των πτυχών της λειτουργικότητας που παρέχει η αναπαράσταση REVE για έναν εικονικό κόσμο, αλλά η επικέντρωση στην φυσική αναπαράσταση των εικονικών κόσμων. Ως εκ τούτου, εστιάζουμε στο *physical aspect* του κόσμου και όχι τόσο στο σημασιολογικό μοντέλο ή στο μοντέλο αλληλεπίδρασης των εικονικών αντικειμένων που συνθέτουν τον κόσμο. Άλλωστε, η διατριβή αυτή έχει ως στόχο να αποτελέσει μια βάση για επέκταση της τρέχουσας λειτουργικότητάς και ενδεχόμενη βελτιστοποίησή της.

Η εργασία αυτή πραγματοποιήθηκε σε συνεργασία με την συνάδελφο Έννα Τσελεμέγκου. Η δική μου συνεισφορά επικεντρώθηκε στο πρωταρχικό κομμάτι της εφαρμογής, δηλαδή της δυνατότητας επιλογής ενός αντικειμένου για την συμμετοχή του σε έναν εικονικό κόσμο, ενώ η συνάδερφος Έννα Τσελεμέγκου επικεντρώθηκε στο κομμάτι της μετατροπής αυτού του αντικειμένου σε *item* και την σχεδίαση του εικονικού κόσμου βάσει της αναπαράστασης REVE.

## Κεφάλαιο 1 - Εισαγωγή Στην Εικονική Πραγματικότητα

Μετά την αλματώδη εξέλιξη που σημειώθηκε στην τεχνολογία του υλικού των υπολογιστών, σημειώνονται αντίστοιχα αποτελέσματα και στην τεχνολογία λογισμικού. Οι ραγδαίες αυτές αλλαγές που συμβαίνουν, δεν μπορούν να αφήσουν ανεπηρέαστο τον τομέα της εκπαίδευσης, όπου δημιουργείται η ανάγκη για νέες τεχνολογίες, οι οποίες υποστηρίζουν νέα περιβάλλοντα μάθησης. Στην εποχή μας δίνεται ιδιαίτερη έμφαση στον τρόπο αλληλεπίδρασης του χρήστη με το υπολογιστικό σύστημα. Επιπλέον, τα εκπαιδευτικά παραδείγματα υποστηρίζουν ότι η γνώση οικοδομείται για αυτό και οι μαθητές είναι υποχρεωμένοι να ασχολούνται περισσότερο με τη μαθησιακή διαδικασία. Συνεπώς, τα περιβάλλοντα που ευνοούν τους μαθητές και που είναι σε θέση να τα διαχειριστούν και να τα εξερευνήσουν, είναι περισσότερο αποτελεσματικά στην επικοινωνιακή μάθηση. Η εικονική πραγματικότητα και κατά συνέπεια και τα εικονικά περιβάλλοντα αποτελούν κύριο εκφραστή της νέας αυτής τάσης.

### 1.1 Τι Είναι Εικονική Πραγματικότητα

Το τι είναι «εικονική πραγματικότητα» δεν μπορούμε να το ορίσουμε και να το εξηγήσουμε. Δεν υπάρχει κάποιος συγκεκριμένος και αυστηρός ορισμός για τον όρο Εικονική Πραγματικότητα, παρακάτω όμως κάνουμε μια προσπάθεια να δώσουμε έναν απλουστευμένο ορισμό. Ο ίδιος ο όρος βέβαια είναι αντιφατικός και οδηγεί σε παρεξηγήσεις και σε πολύωρες φιλοσοφικές συζητήσεις.

Η Εικονική Πραγματικότητα (αγγλ. Virtual Reality (VR)) είναι η προσομοίωση ενός περιβάλλοντος από έναν υπολογιστή. Ο όρος «εικονική πραγματικότητα» χρησιμοποιείται όλο και πιο συχνά στο πλαίσιο της γενικότερης τάσης προς την τρισδιάστατη απεικόνιση. Η εικονική πραγματικότητα ορίζει, γενικά, έναν τεχνητό, τρισδιάστατο ή δισδιάστατο κόσμο με προσθήκη κίνησης μέσω του υπολογιστή, ο οποίος εξυπηρετεί σκοπούς ψυχαγωγίας, εκπαίδευσης και εφαρμογών. Για τη ρεαλιστική βίωση αυτού του ηλεκτρονικού κόσμου απαιτούνται μερικές φορές βοηθητικά μέσα, όπως γυαλιά 3D ή κάποιο γάντι δεδομένων [1].

Η Εικονική Πραγματικότητας χρονολογείται από το 1965, με το δημοφιλές άρθρο της εποχής «The Ultimate Display» του επιστήμονα Ivan Sutherland. Στο εν λόγω άρθρο, ο Sutherland εισήγαγε τις βασικές έννοιες της εμπύθισης σε έναν προσομοιωμένο κόσμο όπου υπάρχουν απόλυτες αισθητηριακές εισοδοι και έξοδοι. Όπως αναλύθηκε και από τον ίδιο:

*«The screen is a window through which one sees a virtual world. The challenge is to make that world look real, act real, sound real, feel real» [2].*

Σε ελεύθερη μετάφραση, «η οθόνη είναι ένα παράθυρο μέσα από το οποίο κάποιος βλέπει έναν εικονικό κόσμο. Η πρόκληση είναι να κάνεις αυτόν τον κόσμο να μοιάζει πραγματικός, να δρα σαν πραγματικός, να ακούγεται πραγματικός, να αισθάνεται πραγματικός».

Ωστόσο, ο όρος Εικονική Πραγματικότητα χρησιμοποιήθηκε πρώτη φορά το 1989 από τον Jaron Larnier (Τζάρον Λέινιερ), ιδρυτή της εταιρίας VPL (Virtual Programming Languages) Research, που ήταν πρωτοπόρα στα συστήματα εικονικής πραγματικότητας. Ο ορισμός που έδωσε ήταν ο εξής: «*Η Εικονική Πραγματικότητα είναι ένα αλληλεπιδραστικό, τρισδιάστατο περιβάλλον, φτιαγμένο από υπολογιστή, στο οποίο μπορεί κάποιος να εμπυθιστεί*» [3].

Η Εικονική Πραγματικότητα χρησιμοποιεί ηλεκτρονικούς υπολογιστές, για να δημιουργήσει και να προσομοιώσει υπαρκτά ή μη περιβάλλοντα, από τα οποία ο χρήστης έχει την ψευδαίσθηση ότι περιβάλλεται και στα οποία μπορεί να κινηθεί ελεύθερα, αλληλεπιδρώντας παράλληλα με τα αντικείμενα που περιλαμβάνουν, όπως θα έκανε και στον πραγματικό κόσμο [4].

Για να είναι όσο πιο πετυχημένη γίνεται η εμπύθιση ενός χρήστη σε ένα περιβάλλον Εικονικής Πραγματικότητας, είναι σημαντικό να απομονωθεί ο χρήστης και οι αισθήσεις του από το πραγματικό κόσμο, επικαλύπτοντας τα ερεθίσματα του πραγματικού κόσμου με αντίστοιχα εικονικά, φτιαγμένα από το σύστημα της Εικονικής Πραγματικότητας. Από τις πέντε (ή μήπως εφτά) αισθήσεις, οι πιο σημαντικές κατά φθίνουσα σειρά είναι η όραση, η ακοή και η αφή. Έτσι είναι πρωταρχικής σημασίας ένα σύστημα Εικονικής Πραγματικότητας να παρέχει

στερεοσκοπική εικόνα, δηλαδή δύο εικόνες από διαφορετική οπτική γωνία, μία για κάθε μάτι του χρήστη, έτσι ώστε να δημιουργηθεί η αίσθηση του βάθους στο χώρο. Παράλληλα η ύπαρξη στερεοσκοπικού ήχου βοηθάει το χρήστη να κατανοεί τι γίνεται γύρω του στον εικονικό χώρο που τον περιβάλλει με πολύ φυσικό τρόπο, ενώ ταυτόχρονα αποκλείει τον χρήστη από τους ήχους του πραγματικού κόσμου, οι οποίοι θα μπορούσαν να καταστρέψουν την εικονική του εμπειρία. Τέλος η αφή, μπορεί να χρησιμοποιηθεί με κατάλληλες συσκευές είτε για να μπορεί ο χρήστης να νιώθει τον κόσμο, είτε για να καθοδηγήσουμε το χρήστη διευκολύνοντάς τον στην εκτέλεση κάποιων συγκεκριμένων ενεργειών. Αν όλα τα παραπάνω συνδυαστούν και με την ανίχνευση των κινήσεων του χρήστη με κατάλληλες συσκευές ανίχνευσης, έτσι ώστε το εικονικό περιβάλλον να συμπεριφέρεται όπως και το πραγματικό, τότε η όλη εμπειρία που θα αποκτήσει ο χρήστης μπορεί να είναι άκρως ρεαλιστική [4].

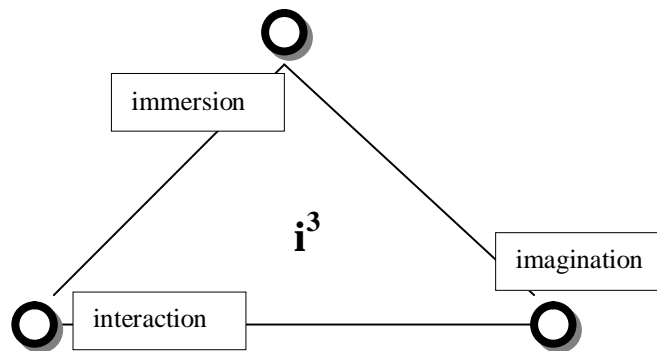
## 1.2 Η Εικονική Πραγματικότητα Στα Γραφικά 3D

Ιδιαίτερα δημοφιλής είναι ο όρος «εικονική πραγματικότητα» στον τομέα των τρισδιάστατων παιχνιδιών υπολογιστή και των τρισδιάστατων ταινιών. Ενώ για την προβολή ταινιών 3D απαιτείται κατάλληλη τηλεόραση με αυτόματη στερεοσκοπική λειτουργία ή πρόσθετα γυαλιά 3D, στα τρισδιάστατα παιχνίδια υπολογιστή, η εικονική πραγματικότητα επιτυγχάνεται μέσω δαπανηρών λύσεων λογισμικού, όπως το DirectX. Το DirectX βοηθά στην απεικόνιση «ζωντανών» σκηνών, όπως δέντρων που λικνίζονται στον άνεμο, λεπτομερών επιδερμίδων εικονικών ζώων ή υδάτινων επιφανειών όπου καθρεφτίζεται το περιβάλλον.

Η εικονική πραγματικότητα χρησιμοποιείται, επίσης, και στην αρχιτεκτονική. Με τη βοήθεια ειδικού λογισμικού CAD ή 3D, οι αρχιτέκτονες σχεδιάζουν τρισδιάστατα μοντέλα των κτιρίων τους στον υπολογιστή, ενώ μπορεί κανείς να περιηγηθεί εικονικά σε αυτά μέσω του υπολογιστή και να τα εξετάσει [5].

## 1.3 Τεχνολογίες Αναπαράστασης

Η Εικονική Πραγματικότητα αποτελεί ουσιαστικά ένα interface υψηλού επιπέδου που περιλαμβάνει προσομοιώσεις πραγματικού χρόνου σε τρισδιάστατο χώρο και αλληλεπιδράσεις μέσα από πολλά κανάλια αισθήσεων. Αυτό έχει σαν αποτέλεσμα η εικονική πραγματικότητα να μπορεί να περιγραφεί με τα τρία I, **immersion** (εμβύθιση), **interaction** (αλληλεπίδραση), **imagination** (φαντασία). Θεωρούμενη ως ένα υπολογιστικό σύστημα, η βασική διάκρισή της από τα συμβατικά είναι ότι θέτει τον άνθρωπο στο κέντρο και οργανώνεται γύρω από τις αισθήσεις του. Είναι εφικτό να περιγραφεί ως ένα τρίγωνο με κορυφές τις τρεις θεμέλιες πτυχές της Καταβύθισης ή εμβύθισης, της φαντασίας και τέλος της αλληλεπίδρασης [6].



**Εικόνα 1: Το Εικονικό Περιβάλλον ως 3I**

Η τεχνολογία αναπαράστασης ενός εικονικού περιβάλλοντος έρχεται αντιμέτωπη με τρία διαφορετικά ζητήματα. Πρώτο ζήτημα είναι η αντίληψη για το σχήμα και την κίνηση των χρηστών και των αντικειμένων. Έπειτα συναντάμε την αλληλεπίδραση αυτών μεταξύ τους βάσει φυσικών κανόνων και τέλος το τρίτο ζήτημα έχει να κάνει με την έκταση και το χαρακτήρα του αναπυσσόμενου περιβάλλοντος. Μια επιτυχημένη περιβαλλοντική προσομοίωση πρέπει να παρέχει κατάλληλα κανάλια επικοινωνίας για να ανταποκρίνεται σε αυτές τις λειτουργίες.

Για την επίλυση αυτών των ζητημάτων πρέπει να συνδυαστούν τρία διαφορετικά είδη hardware: α) αισθητήρες για να ανιχνεύουν την κίνηση του χρήστη, β) επιδραστής, για να εφαρμόζουν τις ενέργειες του χειριστή και γ) ειδικού σκοπού υλικό που συνδέει τους αισθητήρες και τους επιδραστής για να παράγει αισθητηριακές εμπειρίες παρόμοιες με αυτές σε ένα φυσικό περιβάλλον. Η σύνδεση αυτή στο εικονικό περιβάλλον επιτυγχάνεται μέσω ενός υπολογιστή προσομοίωσης.

Τέσσερις τεχνολογίες είναι ουσιαστικές για τα συστήματα εικονικής πραγματικότητας [6]:

1. Οι οπτικές (και ακουστικές και απτικές) οθόνες αναπαράστασης που εμβυθίζουν το χρήστη σε έναν εικονικό κόσμο και αποκόβουν αντικρουόμενες αισθητηριακές εμπειρίες από τον πραγματικό κόσμο.

Τα μοναδικά χαρακτηριστικά της εμβυθισμένης ΕΠ περιγράφονται περιληπτικά ακολούθως [7]:

- Θέαση, η οποία γίνεται με την κίνηση του κεφαλιού, παρέχει μια φυσική διεπαφή για πλοήγηση στον τρισδιάστατο χώρο και επιτρέπει δυνατότητες όπως κοίταγμα τριγύρω, περίπατος, ακόμα και αεροπορική πορεία (flythrough) στα εικονικά περιβάλλοντα.
- Στερεοσκοπική θέαση αυξάνει την αίσθηση του βάθους και του χώρου.
- Ο εικονικός κόσμος αναπαρίσταται σε πλήρη αναλογία και συσχετίζεται με τις ανθρώπινες αναλογίες.
- Ρεαλιστικές αλληλεπιδράσεις με εικονικά αντικείμενα μέσω γαντιών και παρόμοιων συσκευών επιτρέπουν στον χειρισμό και τον έλεγχο των εικονικών κόσμων.
- Η πειστική αυταπάτη της πλήρους εμβύθισης στον εικονικό κόσμο μπορεί να αυξηθεί με ακουστικές, απτικές και άλλες μη οπτικές τεχνολογίες.
- Δικτυακές εφαρμογές επιτρέπουν διαμοιραζόμενα εικονικά περιβάλλοντα.

Την αίσθηση αυτή δίνουν ειδικές συσκευές υλικού (hardware) εικονικής πραγματικότητας, οι οποίες αποτελούν **συσκευές εξόδου** και μπορούν να διακριθούν στις παρακάτω κατηγορίες:

- *Head Mounted Displays (HMDs)*: Πρόκειται για κράνη που ο χρήστης φοράει στο κεφάλι και απομονώνουν την οπτική επαφή με τον πραγματικό κόσμο. Τα κράνη αυτά διαθέτουν δυο μικροσκοπικές στερεοσκοπικές οθόνες (μια για κάθε μάτι), που προβάλλουν τις κινούμενες εικόνες του εικονικού κόσμου. Στα κράνη συνδέονται αισθητήρες κίνησης (motion trackers) που συλλέγουν τις κινήσεις του χρήστη και σε πραγματικό χρόνο προσαρμόζουν την απεικόνιση των οθονών. Έτσι, ο χρήστης μπορεί να εξερευνήσει τον κόσμο, αλλάζοντας οπτικές γωνίες, βασισμένες στην περιστροφή του κεφαλιού [8].



**Εικόνα 2: Παράδειγμα 1 Head Mounted Displays**



**Εικόνα 3: Παράδειγμα 2 Head Mounted Displays**

- *Computer Assisted Virtual Environment (CAVE)*: Πρόκειται για ένα κλειστό κυβικό δωμάτιο, όπου στους τοίχους και το δάπεδο προβάλλονται στερεοσκοπικές εικόνες. Το πάτωμα μπορεί να είναι μια οθόνη προβολής προς τα κάτω, μια επιδαπέδια επιφάνεια προβολής ή μια επίπεδη οθόνη. Τα συστήματα προβολής είναι υψηλής ανάλυσης χάρη στην μικρή απόσταση θέασης, η οποία απαιτεί λίγα εικονοστοιχεία για να διατηρήσει τη ψευδαίσθηση της πραγματικότητας. Ο χρήστης (ή χρήστες) φοράει τρισδιάστατα γυαλιά μέσα στο δωμάτιο αυτό για να δει τα 3D γραφικά που δημιουργούνται από το CAVE και μπορεί να εξερευνήσει τα αντικείμενα που υπάρχουν στο περιβάλλον και να περιπλανηθεί στο χώρο, ενώ αισθητήρες κίνησης συνεχώς αναπροσαρμόζουν τη στερεοσκοπική προβολή του διευθύνοντος ατόμου [8].

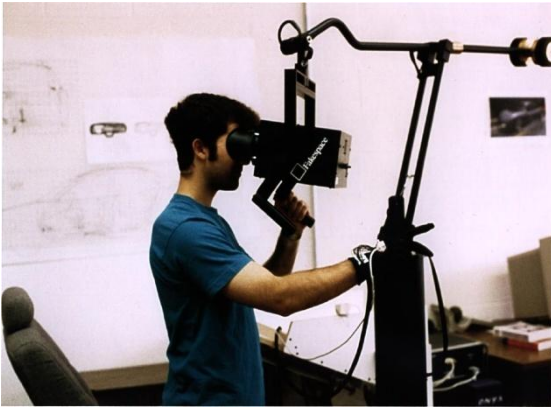


**Εικόνα 4: Παράδειγμα Cave**

- Η πανκατευθυντική διοπτρική οθόνη (*Binocular Omni-directional monitor - BOOM*): Οι οθόνες και το οπτικό σύστημα βρίσκονται σε ένα κουτί το οποίο τοποθετείται σε ένα βραχίονα πολλαπλών συνδέσμων. Ο χρήστης βλέπει τον εικονικό κόσμο κοιτώντας μέσα στο κουτί και μπορεί να καθοδηγήσει το κουτί σε οποιαδήποτε θέση μέσα στον όγκο λειτουργίας της συσκευής. Οι αισθητήρες κίνησης βρίσκονται στους συνδέσμους του βραχίονα που κρατάει το κουτί [8].



**Εικόνα 5: Παράδειγμα 1 χρήσης του BOOM**



Εικόνα 6: Παράδειγμα 2 χρήσης του BOOM

- *Πανοραμικές οθόνες (panoramic displays):* Μια ή περισσότερες οθόνες τοποθετούνται εναλλάξ στον ίδιο χώρο σε πανοραμική θέση, όπου ένας χρήστης καθοδηγεί την οπτική γωνία. Χρησιμεύει κυρίως όταν υπάρχουν πολλαπλοί χρήστες στον ίδιο χώρο [8].



Εικόνα 7: Παράδειγμα 1 πανοραμικών οθονών



Εικόνα 8: Παράδειγμα 2 πανοραμικών οθονών

- *Virtu-Sphere:* Η ομώνυμη εταιρία έχει δημιουργήσει μια μπάλα στο μέγεθος ενός ανθρώπου και συνδυάζει αισθητήρες κίνησης και HDM για μια μοναδική εμπειρία εμπύθισης. Ο χρήστης εισέρχεται στη σφαίρα η οποία βρίσκεται στερεωμένη στο έδαφος και φοράει ένα HDM, όπου προβάλλεται ο εικονικός κόσμος. Ο χρήστης μπορεί



να προχωρήσει στη σφαίρα. Ενώ ο χρήστης παραμένει στατικός, η σφαίρα περιστρέφεται αντίστοιχα στις κινήσεις του χρήστη, με δυνατότητα περιστροφής 360°. Αισθητήρες στους τροχούς ενημερώνουν τη CPU για το πώς κινείται ο χρήστης. Αντίστοιχα ελέγχεται η κίνηση και για την ανανέωση του κόσμου μέσω του HDM [8].



Εικόνα 9: Παράδειγμα 1 VirtuSphere

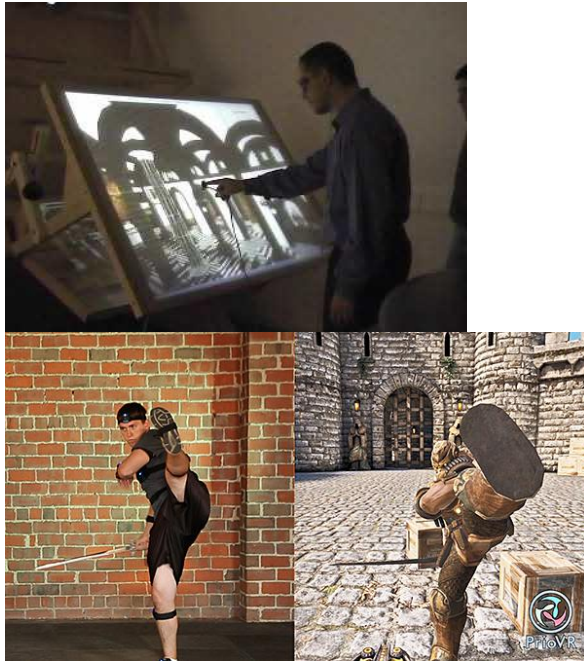


Εικόνα 10: Παράδειγμα 2 VirtuSphere

2. Το σύστημα απόδοσης γραφικών (rendering) που παράγει διαρκώς εναλλασσόμενες εικόνες σε ρυθμό από 20 ως 120 πλαίσια το δευτερόλεπτο [8].
3. Το σύστημα ανίχνευσης (tracking), που αναφέρει διαρκώς τη θέση του κεφαλιού και των άκρων του χειριστή [8].

Για το σύστημα ανίχνευσης χρησιμοποιούνται διάφορες συσκευές εισόδου που έχουν αισθητήρες για να αντιλαμβάνονται την κίνηση του χρήστη και να στέλνουν την απαραίτητη πληροφορία στον υπολογιστή που είναι υπεύθυνος για την εξομοίωση της αλληλεπίδρασης. Τέτοιες συσκευές εισόδου είναι οι παρακάτω:

- **Ανιχνευτές (trackers):** Πρόκειται για αισθητήρες που τοποθετούνται σε συγκεκριμένα σημεία του σώματος (πχ χέρια, πόδια, κεφάλι) ή σε όλο το σώμα και ανιχνεύουν τη θέση του χρήστη, τον προσανατολισμό του ή και τα δυο. Εφαρμόζονται για ανίχνευση κίνησης κεφαλιού, ματιών, προσώπου, χεριών ή και όλου του σώματος, όπως φαίνεται στις παρακάτω εικόνες:



**Εικόνα 11: Ανιχνευτές χεριού και σώματος**

Οι ανιχνευτές μπορεί να είναι μαγνητικοί που είναι και οι πιο συνηθισμένοι, οι οποίοι τοποθετούνται στο αντικείμενο που θέλουμε να παρακολουθήσουμε την κίνησή του, οπτικοί, που κάνουν χρήση στερεοσκοπικών καμερών ή ακουστικοί, όπου οι συσκευές εισόδου είναι μικρόφωνα [8].

- *Γάντια δεδομένων (data gloves):* Πρόκειται για μια ηλεκτρομηχανική συσκευή που θυμίζει ένα κλασικό γάντι χεριού και φοριέται ως τέτοιο. Χρησιμοποιείται για απτικές εφαρμογές, έχοντας αισθητήρες που επιτρέπουν την ανίχνευση της κίνησης του χεριού και των δακτύλων και αντιλαμβάνονται την πίεση, τη στροφορμή, τη γραμμική δύναμη, τη θερμοκρασία και την υφή μιας επιφάνειας, επιτρέποντας έτσι την αλληλεπίδραση με εικονικά αντικείμενα [8].



**Εικόνα 12: Γάντια δεδομένων**

- Άλλες συσκευές εισόδου που επιτρέπουν αλληλεπίδραση με το εικονικό περιβάλλον είναι το κλασικό και το τρισδιάστατο ποντίκι (3Dmouse), η μπάλα ανίχνευσης (trackball), η μπίλια (spaceball) και το ηλεκτρονικό χειριστήριο (joystick) [8].

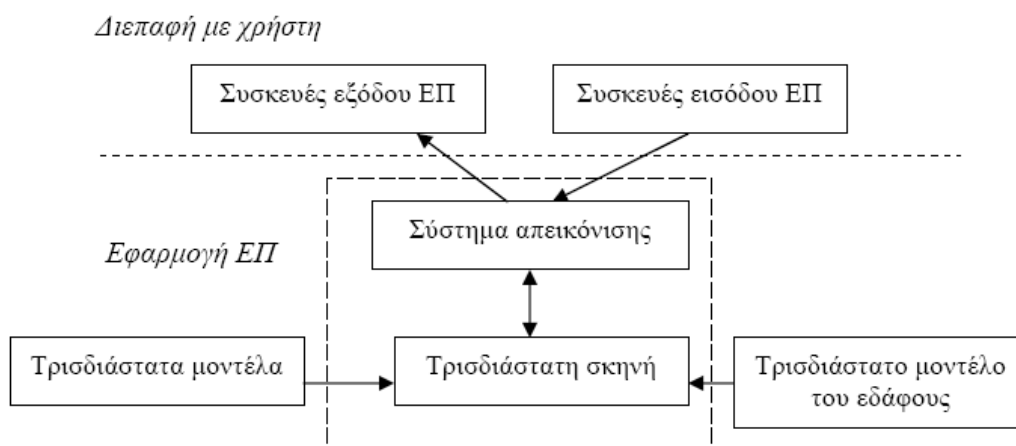


Εικόνα 13: Παράδειγμα trackball (αριστερά), spaceball (κέντρο) και joystick (δεξιά)

4. Η δομή της **βάσης δεδομένων** και το **σύστημα συντήρησης** για την παραγωγή και τη διατήρηση λεπτομερών και ρεαλιστικών μοντέλων του εικονικού κόσμου [8].

#### 1.4 Συστατικά Συστήματος Εικονικής Πραγματικότητας

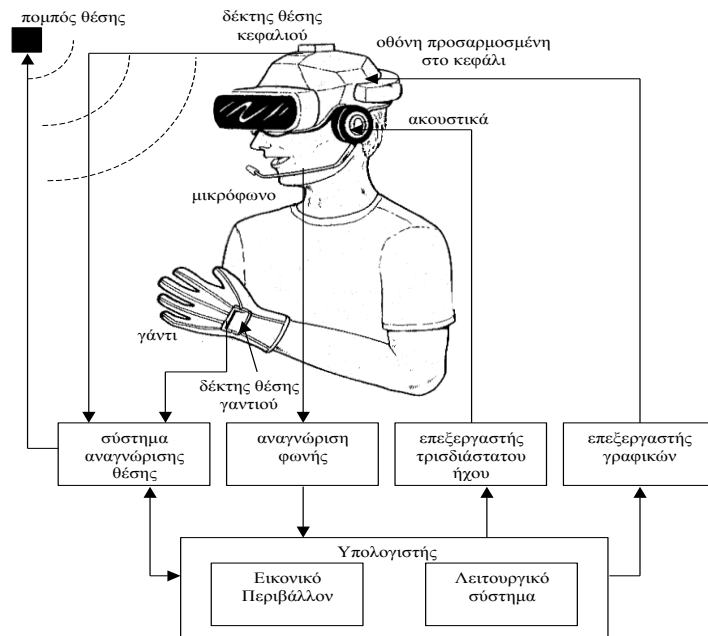
Ένα σύστημα εικονικής πραγματικότητας (ΕΠ) αποτελείται από τα συστατικά που φαίνονται στην παρακάτω εικόνα.



Εικόνα 14: Συστατικά συστήματος εικονικής πραγματικότητας

Το σύστημα απεικόνισης (viewer) και η τρισδιάστατη σκηνή συνδέονται στενά αφού η επιλογή του τρισδιάστατου περιβάλλοντος απεικόνισης ως 3D viewer υποδηλώνει μια τρισδιάστατη υλοποίηση του σκηνικού (3D scene). Η τρισδιάστατη σκηνή λαμβάνει συνεισφορές από ένα τρισδιάστατο μοντέλο του εδάφους και τρισδιάστατες απεικονίσεις των αντικειμένων του πραγματικού κόσμου. Και τα δυο μαζί αποτελούν την τρισδιάστατη μηχανή απεικόνισης (3D player engine). Το μοντέλο εδάφους αποτελεί μια γεωγραφική βάση δεδομένων του εδάφους σε τρισδιάστατη μορφή. Τα Τρισδιάστατα μοντέλα του πραγματικού κόσμου αποτελούν την δομή της βάσης δεδομένων και το σύστημα συντήρησης για την παραγωγή και τη διατήρηση λεπτομερών και ρεαλιστικών μοντέλων του εικονικού κόσμου. Οι συσκευές εισόδου ΕΠ είναι συσκευές που έχουν αισθητήρες για να αντιλαμβάνονται την κίνηση του χρήστη και να στέλνουν

την απαραίτητη πληροφορία στον υπολογιστή που είναι υπεύθυνος για την εξομίωση της αλληλεπίδρασης. Τέλος οι συσκευές εξόδου ή απεικόνισης ΕΠ αναφέρονται στο σύστημα απόδοσης γραφικών (rendering) που παράγει διαρκώς εναλλασσόμενες εικόνες σε ρυθμό από 20 ως 120 πλαίσια το δευτερόλεπτο [9][10].



Εικόνα 15: Γενικό σύστημα Εικονικού Περιβάλλοντος με τα επιμέρους του όργανα

## 1.5 Η Εικονική Πραγματικότητα Ως Πρακτική Εφαρμογή

Πολλοί ερευνητές και εκπαιδευτικοί επαγγελματίες θεωρούν ότι η τεχνολογία της ΕΠ προσφέρει ισχυρά οφέλη που μπορούν να υποστηρίξουν την εκπαίδευση. Μπορεί να επηρεάσει θετικά τη μαθησιακή διαδικασία γιατί παρέχει ισχυρή αλληλεπίδραση, άμεση ανταπόκριση του συστήματος στις ενέργειες του χρήστη και ελευθερία των κινήσεων στην εικονική κατάσταση. Για μερικούς, η δυνατότητα της VR να διευκολύνει τις κονστρακτιβιστικές δραστηριότητες εκμάθησης είναι το βασικό ζήτημα. Η τεχνολογία της ΕΠ θα μπορούσε να είναι ένα ισχυρό εργαλείο για την εκπαίδευση βασισμένη στις immersive και δυναμικές ιδιότητές της.

Όσον αφορά τον τομέα της εξομίωσης, η εικονική πραγματικότητα εξυπηρετεί κάποιους πολύ πρακτικούς σκοπούς εφαρμογών. Μια μορφή εικονικής πραγματικότητας είναι π.χ. ο εξομωτής πτήσεων, όπου εκπαιδεύονται οι πιλότοι. Εν προκειμένω, η εικονική πραγματικότητα είναι ένας συνδυασμός της κινούμενης μέσω του υπολογιστή πραγματικότητας στην οθόνη, εξομιώνοντας το παράθυρο του πιλοτηρίου, και του θαλάμου εκπαίδευσης, ο οποίος είναι διαμορφωμένος σαν πιλοτήριο. Οι αλεξιπτωτιστές του αμερικανικού ναυτικού εκπαιδεύονται με ένα πραγματικό αλεξίπτωτο-χαλινάρι, το οποίο κρέμεται, και φορούν ειδικά γυαλιά με επανομαζόμενη οθόνη «Head Mounted» που μεταδίδει εικονικά το άλμα από το αεροσκάφος.

Η προσφορά της εικονικής πραγματικότητας στην εκπαιδευτική διαδικασία φαίνεται από τις παρακάτω δυνατότητές τους: εξερεύνηση υπαρκτών αντικειμένων και χώρων στους οποίους δεν υπάρχει πρόσβαση από τους μαθητές δημιουργία και χειρισμός αφηρημένων αναπαραστάσεων μελέτη πραγματικών αντικειμένων τα οποία είναι αδύνατο να κατανοηθούν διαφορετικά εξαιτίας του μεγέθους της θέσης και των ιδιοτήτων τους δημιουργία περιβαλλόντων και αντικειμένων τα οποία έχουν διαφορετικές από τις γνωστές ιδιότητες αλληλεπίδραση με πραγματικούς ανθρώπους σε μακρινές φυσικές θέσεις ή φανταστικούς τόπους με πραγματικούς ή μη τρόπους.

Οι σχεδιαστές και οι ερευνητές χρησιμοποιούν ορισμένες φορές ειδικά γυαλιά ή ακόμη και κάποιο πρόσθετο γάντι δεδομένων, για να σχεδιάσουν, να αναπτύξουν και να κατανοήσουν επακριβώς τους εικονικούς κόσμους. Χάρη σε λειτουργίες όπως το «Tactile Feedback» και το «Force Feedback» εξομοιώνουν την πραγματική ψηλάφηση εικονικών αντικειμένων και των σχετικών αναδράσεων δύναμης.

## 1.6 Το Μέλλον Της Εικονικής Πραγματικότητας

Σε αντίθεση με το παρελθόν, το οποίο ήταν γεμάτο υποσχέσεις οι οποίες τελικά ποτέ δεν πραγματοποιήθηκαν γιατί οι τεχνολογίες και οι συνθήκες που επικρατούσαν δεν ήταν αρκετά ώριμες, το μέλλον της Εικονικής Πραγματικότητας υπόσχεται μια συνεχή ανάπτυξη σε πιο στέρεες και λιγότερο ουτοπικές βάσεις. Πλέον η βιομηχανία αρχίζει να βρίσκει πρακτικές εφαρμογές στη χρήση των Εικονικών Περιβαλλόντων, όπως είναι η σχεδίαση αυτοκινήτων ή η εκπαίδευση αστροναυτών, και μαζί με τα διάφορα κρατικά ερευνητικά προγράμματα προσφέρουν μια σταθερή χρηματοδότηση για την έρευνα και την ανάπτυξη τέτοιων συστημάτων, καθώς και την εύρεση νέων εφαρμογών τους. Μάλιστα τελευταία γίνονται προσπάθειες να συγκεντρωθεί όλη η γνώση που έχει συσσωρευτεί διάσπαρτη σε ερευνητικά ιδρύματα και εταιρείες που ασχολούνται με την Εικονική Πραγματικότητα, με σκοπό την κατανόηση των αναγκών τέτοιων περιβαλλόντων καθώς και την ανάπτυξη οδηγιών για τη σχεδίαση και υλοποίησή τους. Τέλος δεν πρέπει να ξεχνάμε ότι η συνεχής ανάπτυξη της τεχνολογίας, δίνει τη δυνατότητα για αναβάθμιση της ποιότητας της εμπύθισης που μπορεί να γίνει αντιληπτή από το χρήστη, μέσω της βελτίωσης των συσκευών εισόδου και εξόδου, ενώ παράλληλα μειώνεται και το κόστος αυτών των συσκευών, που τουλάχιστον μέχρι τώρα είναι απαγορευτικό για το μέσο χρήστη. Έτσι θα λέγαμε ότι δεν θα αργήσει η μέρα που η Εικονική Πραγματικότητα θα βρεθεί σε κάθε σπίτι, ιδιαίτερα αν υποστηριχθεί από τον τομέα της ψυχαγωγίας και των παιχνιδιών.

Συμπερασματικά, η εικονική πραγματικότητα έχει σημαντικές εκπαιδευτικές και παιδαγωγικές εφαρμογές οι οποίες μπορούν να επεκταθούν σε μεγάλο εύρος διδακτικών και μαθησιακών χώρων ανοίγοντας έτσι νέους δρόμους στην εκπαίδευση και την κατάρτιση. Επιτρέπουν να προσομοιώσου με νέα, δικής μας επινόησης, περιβάλλοντα. Τα περιβάλλοντα αυτά δεν λειτουργούν πλέον κάτω από τους νόμους της γνωστής φυσικής και της κοινής πραγματικότητας. Συνεπώς μπορούν να μετασχηματισθούν αρκετά εύκολα τις περισσότερες φορές από τη δραστηριότητα των εξερευνητών τους.

Τελευταία στην επιστημονική κοινότητα αποφεύγεται η χρήση του όρου Εικονική Πραγματικότητα λόγω της αντιφατικότητάς του και χρησιμοποιείται ο όρος Εικονικό Περιβάλλον, Virtual Environment στα αγγλικά, (αγγλική συντομογραφία VE).

## Κεφάλαιο 2 - Εικονικά Περιβάλλοντα

Τα εικονικά περιβάλλοντα είναι μια διαρκώς αναπτυσσόμενη περιοχή με πολλές και ενδιαφέρουσες εφαρμογές σε διάφορους τομείς, όπως η εκπαίδευση, η διασκέδαση και η προσομοίωση. Ένα σημαντικό τμήμα των εικονικών περιβαλλόντων που ενισχύει την αυτονομία και την αλληλεπιδραστικότητά τους είναι οι εικονικοί πράκτορες, δηλαδή οι αυτόνομες οντότητες του περιβάλλοντος που είναι σε θέση να αντιλαμβάνονται τον κόσμο γύρω τους και να εκτελούν ενέργειες ώστε να επιτύχουν τους στόχους τους.

### 2.1 Ανάπτυξη Και Κατηγορίες Εικονικών Περιβαλλόντων

Ένα εικονικό περιβάλλον αποτελείται από περιεχόμενο (αντικείμενα και δράστες (actors)), γεωμετρία και δυναμική, με ένα εγωκεντρικό πλαίσιο αναφοράς, που περιλαμβάνει την αντίληψη των αντικειμένων σε βάθος και που εγείρει διάφορες αισθήσεις ταυτόχρονα καθώς και την αλληλεπίδραση ενός ή παραπάνω χρηστών με αυτό [11].

Οι ραγδαίοι ρυθμοί ανάπτυξης των προσωπικών υπολογιστών την τελευταία δεκαετία έχουν δώσει στους απλούς χρήστες τη δυνατότητα να έρθουν σε επαφή με εφαρμογές γραφικών, που χρησιμοποιούν πολύπλοκα τρισδιάστατα μοντέλα, τα οποία κάποτε χρειαζόταν πανίσχυρα μηχανήματα για να τρέξουν σε πραγματικό χρόνο. Σήμερα τα τρισδιάστατα γραφικά βρίσκουν όλο και συχνότερη εφαρμογή σε χώρους, όπως η διασκέδαση, η εκπαίδευση και η προσομοίωση, καθώς παρέχουν στο χρήστη μια πιο ρεαλιστική και ολοκληρωμένη άποψη των αντικειμένων και ενισχύουν με αυτόν τον τρόπο τις δυνατότητες αλληλεπίδρασής του αλλά και το ενδιαφέρον του για την εφαρμογή.

Ανάλογα το εξειδικευμένο υλικό που χρησιμοποιείται για την επικοινωνία με το χρήστη, πράγμα που καθορίζει και τον τρόπο αλληλεπίδρασής του με το εικονικό περιβάλλον, μπορούμε να διακρίνουμε διάφορες κατηγορίες εικονικών περιβαλλόντων. Για παράδειγμα, τα Περιβάλλοντα Εμβύθισης (immersive environments) είναι εικονικά περιβάλλοντα στα οποία οι χρήστες είναι εφοδιασμένοι με το κατάλληλο υλικό, πχ. χρήση HDMs, ώστε να δέχονται ερεθίσματα μόνο από το συνθετικό περιβάλλον και όχι από τον πραγματικό κόσμο. Στα Ενισχυμένα Εικονικά Περιβάλλοντα (augmented environment) η διαφορά τους έγκειται στο ότι τα εικονικά αντικείμενα προστίθενται πάνω στην άποψη του πραγματικού κόσμου, εμπλουτίζοντας στην ουσία το πραγματικό περιβάλλον του χρήστη με το εικονικό.

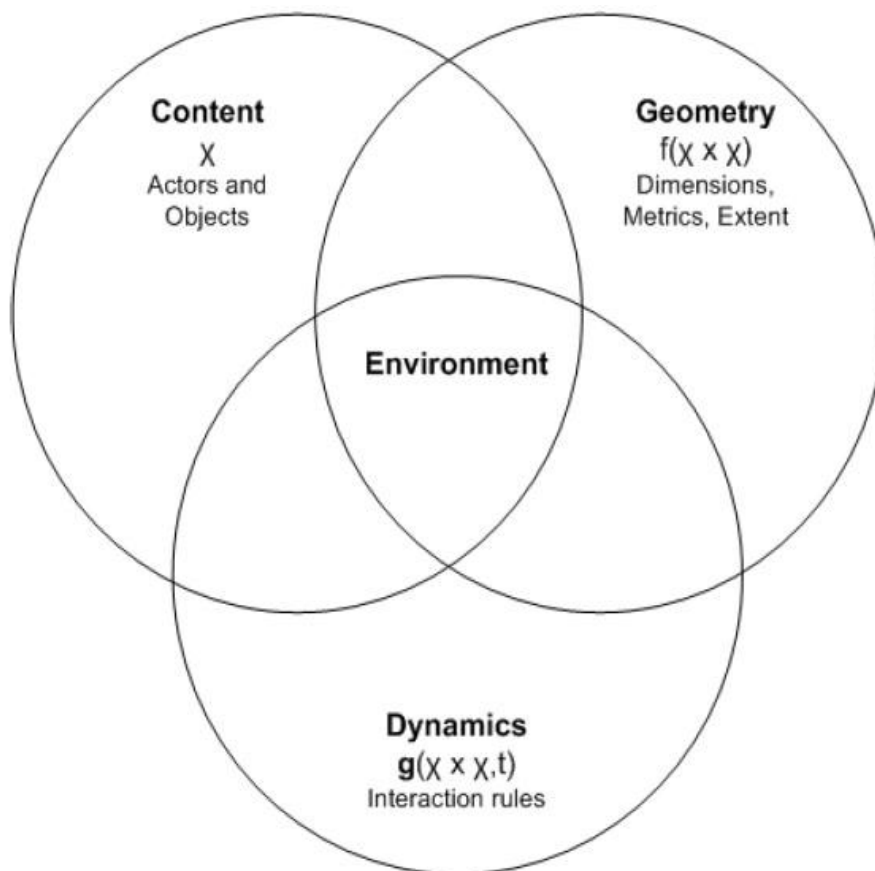
Από την άλλη, όταν ο τεχνητός κόσμος προβάλλεται ολόκληρος πάνω σε μια ή περισσότερες επιφάνειες του πραγματικού κόσμου (π.χ. τοίχος, τραπέζι ή δωμάτιο) ή χρησιμοποιούνται πολλαπλές οθόνες που κυκλώνουν τον χρήστη, όπως στην περίπτωση του CAVE, τότε πρόκειται για Περιβάλλοντα Προβολής (projected environments). Ακόμη, υπάρχει και η πιο συνήθης και πολυχρησιμοποιημένη περίπτωση εικονικού περιβάλλοντος που είναι το Περιβάλλον Οθόνης (desktop environment), όπου ένας τρισδιάστατος εικονικός κόσμος απεικονίζεται στην δισδιάστατη οθόνη του υπολογιστή. Σε αυτή την περίπτωση μπορεί να χρησιμοποιούνται περιφερειακές συσκευές πλοήγησης στον τρισδιάστατο εικονικό χώρο και να γίνεται χρήση στερεοσκοπικών γυαλιών ή κράνους [10].

Ακόμα, ανάλογα τους χρήστες που πλοηγούνται σε ένα εικονικό περιβάλλον, μπορούμε να τα διακρίνουμε σε συστήματα για έναν χρήστη (single-user VEs) ή σε πολυχρηστικά καταμεμημένα εικονικά περιβάλλοντα (multi-user, collaborative, distributed VEs), που επιτρέπουν σε μια ομάδα διασκορπισμένων χρηστών να αλληλεπιδρούν σε πραγματικό χρόνο [10].

### 2.2 Λειτουργικά Χαρακτηριστικά Εικονικού Περιβάλλοντος

Μια πολύ εύστοχη ανάλυση ενός VE (Εικονικό Περιβάλλον) στα λειτουργικά στοιχεία από τα οποία αποτελείται πραγματοποιήθηκε από τον, σύμφωνα με την οποία ένα εικονικό περιβάλλον συνίσταται σε:

- **Περιεχόμενο:** Τα αντικείμενα (objects) και τα ενεργά ή δρώντα στοιχεία (actors) τα οποία μπορούν να θεωρηθούν και αυτά σαν αντικείμενα, αλλά σαν έχοντα την δυνατότητα να ξεκινούν από μόνα τους αλληλεπιδράσεις με άλλα αντικείμενα του VE. Ένα τέτοιο στοιχείο είναι ο ίδιος ο χρήστης που αντιπροσωπεύεται στο VE από τη δική του οπτική άποψη (viewpoint) του περιβάλλοντος.
- **Γεωμετρία:** δηλαδή την περιγραφή του πεδίου όπου εξελίσσεται η αλληλεπίδραση.
- **Δυναμικές:** δηλαδή τους κανόνες της αλληλεπίδρασης ανάμεσα στα συστατικά του περιβάλλοντος, οι οποίοι περιγράφουν την συμπεριφορά των συστατικών αυτών καθώς ανταλλάσσουν ενέργεια ή πληροφορία [11].



Εικόνα 16: Λειτουργικά χαρακτηριστικά ενός Εικονικού Περιβάλλοντος

### 2.3 Εφαρμογές Εικονικών Περιβαλλόντων

Οι εφαρμογές και η χρήση των εικονικών περιβαλλόντων είναι ευρεία και επεκτείνεται σε πληθώρα επιστημονικών κλάδων, γεγονός που αναδεικνύει και τη σημασία τους ως σύγχρονου μέσου αλληλεπίδρασης ανθρώπου-υπολογιστή. Ένας από τους τομείς που βρίσκουν δημοφιλή εφαρμογή στα εικονικά περιβάλλοντα είναι η εκπαίδευση. Αυτό επιτυγχάνεται γιατί η δυνατότητα πειραματισμού και παρατήρησης των δεδομένων από διαφορετικές σκοπιές και χωρίς τις πραγματικές συνέπειες που θα είχαν λάθη στον φυσικό κόσμο, ενισχύει τη διαδικασία της μάθησης και της κατανόησης. Για παράδειγμα, μέσα από το πρόγραμμα Newton Worlds χρησιμοποιούνται εικονικά περιβάλλοντα για την εκμάθηση της φυσικής ή της χημείας μέσω αλληλεπίδρασης με χημικές δομές [12].



Ένα ακόμα σημαντικό πεδίο που βρίσκουν εφαρμογή τα εικονικά περιβάλλοντα είναι ο στρατός και πιο συγκεκριμένα χρησιμεύουν στην στρατιωτική εκπαίδευση των στρατιωτών προσομοιάζοντας ένα εικονικό πεδίο μάχης. Ένα παράδειγμα είναι το Javelin Virtual Combat System που χρησιμοποιείται στην Αμερική [13].



**Εικόνα 17: ΕΠ για στρατιωτική εκπαίδευση**

Επίσης, η δυνατότητα προσομοίωσης ρεαλιστικών κόσμων βρίσκει μεγάλη χρησιμότητα στον τηλεχειρισμό (teleoperation), ιδιαίτερα σε περιπτώσεις που οι αντίστοιχες ενέργειες σε πραγματικές συνθήκες μπορεί να έχουν μεγάλο κόστος ή και να είναι επικίνδυνες, όπως στην ναυπηγική, σε προσομοιώσεις πτήσης και σε εικονικές χειρουργικές επεμβάσεις.





**Εικόνα 18: ΕΠ προσομοίωσης χειρουργικής επέμβασης**

Αντίστοιχα, τα εικονικά περιβάλλοντα βρίσκουν εφαρμογή και στις επιχειρήσεις, καθώς επιτρέπουν την αναπαράσταση και το χειρισμό ενός μοντέλου, με σκοπό τον έλεγχο και τη βελτίωση της λειτουργικότητάς του πριν ακόμα βγει στο στάδιο της παραγωγής.

Με τη βοήθεια ειδικού λογισμικού CAD (Computer Aided Design) ή 3D, οι αρχιτέκτονες έχουν την ευχέρεια να σχεδιάζουν τρισδιάστατα μοντέλα κτιρίων στον υπολογιστή, στα οποία μπορεί κανείς να περιηγηθεί εικονικά μέσω του υπολογιστή και να τα εξετάσει τόσο στο εσωτερικό τους όσο και στον εξωτερικό περιβάλλοντα χώρο.

Επιπλέον, εικονικά περιβάλλοντα χρησιμοποιούνται στην NASA για την απομακρυσμένη εξερεύνηση πλανητικών επιφανειών. Προσφέρονται τεχνικές επίλυσης προβλημάτων ελέγχου που δημιουργούνται από τη χρονοκαθυστέρηση. Αντίστοιχα σε πεδία της ρομποτικής είναι άκρως χρήσιμα τα εικονικά περιβάλλοντα στην ανάλυση και στην οπτικοποίηση επιστημονικών δεδομένων. Τέλος, βρίσκουν εφαρμογή και στην τέχνη, με την εξέλιξη της διαδραστικής τέχνης και βέβαια, στην ψυχαγωγία, όπου τα τελευταία χρόνια πρωταγωνιστικό ρόλο έχει η βιομηχανία παραγωγής βιντεοπαιχνιδιών. Βάση των παραπάνω και δεδομένου ότι τα εικονικά περιβάλλοντα προσφέρουν δυνατότητες στον χρήστη που υπερβαίνουν τα όρια του φυσικού κόσμου.

Θα μπορούσαμε να διακρίνουμε σε δυο κατηγορίες τους ανθρώπους που ασχολούνται ή ενδιαφέρονται για τα εικονικά περιβάλλοντα: α) σε αυτούς οι οποίοι επιθυμούν να χρησιμοποιήσουν την εν λόγω τεχνολογία για να εξελίξουν το επάγγελμα ή το ενδιαφέρον τους πάνω σε ένα συγκεκριμένο αντικείμενο με τη βοήθεια των εικονικών περιβαλλόντων, και β) σε εκείνους που επιθυμούν να αναπτύξουν και να βελτιώσουν την τεχνολογία ανάπτυξης των εικονικών περιβαλλόντων. Στην συγκεκριμένη διπλωματική διατριβή θα ασχοληθούμε με την ανάπτυξη και βελτίωση της ίδιας της τεχνολογίας αναπαράστασης εικονικών περιβαλλόντων.

## 2.4 Ευφυές Εικονικό Περιβάλλον

Ένα εικονικό περιβάλλον αν είναι στατικό και δεν υφίσταται αλλαγές τότε η εμπειρία εμπύθισης που προκαλεί έχει περιορισμένο ενδιαφέρον, με αποτέλεσμα να μην είναι για μεγάλο χρονικό διάστημα είναι εντυπωσιακό και ελκυστικό. Επομένως, είναι απαραίτητο να δημιουργηθεί ένας αυτοματοποιημένος τρόπος παραγωγής συμπεριφοράς στα εικονικά περιβάλλοντα, ώστε να είναι πιο δυναμικά και ενδιαφέροντα, επεκτείνοντας κατά αυτόν τον τρόπο τις δυνατότητες εφαρμογής τους.

Η ευκαιρία δόθηκε κυρίαρχα με την αύξηση της επεξεργαστικής ισχύος που χρησιμοποιείται για το rendering, καθιστώντας εφικτό ένα κομμάτι αυτής της ισχύος να αφιερωθεί σε θέματα πέραν της ρεαλιστικής απεικόνισης, εγκαινιάζοντας ένα νέο επιστημονικό πεδίο. Αυτό το νέο επιστημονικό πεδίο έγινε γνωστό με τον όρο Ευφυή Εικονικά Περιβάλλοντα.

«Η αυξανόμενη επικάλυψη της τεχνολογίας που αφορά τα τρισδιάστατα διαδραστικά γραφικά περιβάλλοντα σε πραγματικό χρόνο και της τεχνολογίας της Τεχνητής Νοημοσύνης οδήγησε στην ανάπτυξη ενός νέου επιστημονικού τομέα, γνωστού με τον όρο Ευφυή Εικονικά Περιβάλλοντα» [14]. Τα Ευφυή Εικονικά Περιβάλλοντα είναι συνδυασμός έξυπνων τεχνικών και εργαλείων, που ενσωματώθηκαν σε αυτόνομες οντότητες (πράκτορες) μαζί με αποδοτικά μέσα για την γραφική αναπαράστασή τους και την δυνατότητα αλληλεπίδρασής τους» [15].

Πιο απλά, με τον όρο Ευφυή Εικονικά Περιβάλλοντα (Intelligent Virtual Environments) ορίζεται η σύγκλιση δυο διαφορετικών επιστημονικών πεδίων, της Τεχνητής Νοημοσύνης και της Εικονικής Πραγματικότητας, επιτρέποντας ευφυείς οντότητες να αναπαρίστανται και να δρουν αυτόνομα σε ένα εικονικό περιβάλλον. Μπορούμε να συναντήσουμε τον όρο και ως *Ευφυή Συστήματα Εικονικής Πραγματικότητας* (Intelligent Virtual Reality Systems), καθώς η νοημοσύνη ενσωματώνεται στην ίδια την αρχιτεκτονική του συστήματος, υλοποιώντας αλγορίθμους Τεχνητής Νοημοσύνης σε ένα σύστημα εικονικής πραγματικότητας [14].

Υπάρχουν διάφοροι παράγοντες που οδήγησαν στην διασύνδεση αυτών των δυο πεδίων, εκ των οποίων διακρίνουμε τους εξής τρεις: αρχικά η συνεχόμενη αύξηση του μεγέθους υπολογιστικής ισχύος που μπορεί να σηκώσει ένας επιτραπέζιος υπολογιστής. Η αύξηση υπολογιστικής ισχύς όχι μόνο υποστηρίζει έναν υψηλότερο βαθμό ρεαλιστικής απεικόνισης, αλλά αφήνει διαθέσιμη υπολογιστική ισχύ που μπορεί να χρησιμοποιηθεί και για την πρόσθεση νοημοσύνης. Ο δεύτερος παράγοντας σχετίζεται με την εξέλιξη και την διαδεδομένη διαθεσιμότητα λογισμικού για 3D γραφικά και την ανάπτυξη προτύπων για 3D γραφικά, όπως η VRML '97. Και τρίτον, οι τεχνολογίες Τεχνητής Νοημοσύνης, έχουν αναπτυχθεί στο βαθμό που μπορούν να χρησιμοποιηθούν ως μέσο διάδρασης εντός ενός εικονικού περιβάλλοντος [15].

## 2.5 Ελλείψεις Στον Τομέα Των Εικονικών Περιβαλλόντων

Παρά τη χρησιμότητα των εικονικών περιβαλλόντων σε διάφορους τομείς και την πληθώρα έρευνας που έχει πραγματοποιηθεί γύρω από αυτά τις τελευταίες δεκαετίες, ο χώρος παρουσιάζει ακόμα σημαντικές ελλείψεις. Πρώτα απ' όλα, το εξειδικευμένο υλικό που χρησιμοποιείται είναι αφενός υπερβολικά ακριβό για το μέσο χρήστη, αφετέρου όχι ιδιαίτερα αξιόπιστο, με αποτέλεσμα η συνεχής χρήση του για μεγάλο χρονικό διάστημα να προκαλεί δυσφορία. Κατά συνέπεια, ένα μεγάλο μέρος της έρευνας επικεντρώνεται στα εικονικά περιβάλλοντα οθόνης, που φαίνεται να είναι η μόνη προσιτή λύση για το μέσο χρήστη.

Δεύτερον, φαίνεται να υπάρχει έλλειψη γενικού τύπου αρχιτεκτονικών και εργαλείων ανάπτυξης για εικονικά περιβάλλοντα, με αποτέλεσμα να είναι ιδιαίτερα επίπονη και χρονοβόρα η διαδικασία κατασκευής ενός τέτοιου περιβάλλοντος, ενώ απαιτείται και σημαντικό χρονικό διάστημα για τον έλεγχο και την αποσφαλμάτωσή του, ώστε να παρέχει στο χρήστη ένα αξιόπιστο και φιλικό περιβάλλον αλληλεπίδρασης.

Τέλος, στα υπάρχοντα εικονικά περιβάλλοντα παρουσιάζεται έλλειψη αυτονομίας. Σύμφωνα με την ταξινόμηση του [16] οι τρεις διαστάσεις αξιολόγησης ενός εικονικού περιβάλλοντος είναι η αυτονομία (autonomy), δηλαδή η ικανότητα των αντικειμένων να δρουν από μόνα τους, η αλληλεπίδραση (interaction), δηλαδή το ποσοστό στο οποίο το σύστημα παρέχει έλεγχο των αντικειμένων, και η παρουσία (presence), που καθορίζεται από το εύρος και

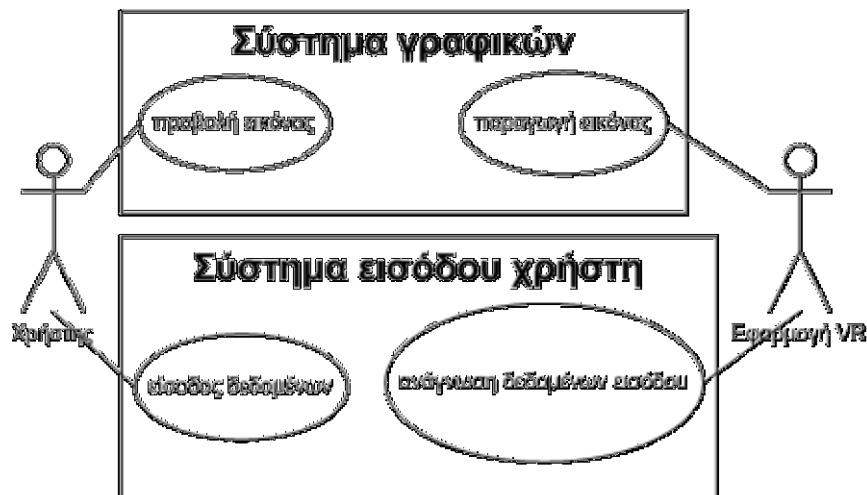
την πιστότητα των αισθητηρίων καναλιών του χρήστη που ελέγχει το περιβάλλον. Ενώ λοιπόν σήμερα οι χρήστες μπορούν να έχουν πολύπλοκες αλληλεπιδράσεις με τα αντικείμενα χρησιμοποιώντας εξειδικευμένο υλικό, όπως τα γάντια δεδομένων (data gloves), και η παρουσία τους είναι ιδιαίτερα ενισχυμένη στα περιβάλλοντα προβολής και εμπύθισης, δε φαίνεται να υπάρχει αντίστοιχη πρόοδος στον τομέα της αυτονομίας. Αυτή η έλλειψη οφείλεται κυρίως στο γεγονός ότι μέχρι πρόσφατα η υπολογιστική ισχύς των μηχανημάτων αφιερωνόταν σχεδόν αποκλειστικά στην απεικόνιση και συνθετική κίνηση του κόσμου, με αποτέλεσμα να μην περισσεύει επιπλέον υπολογιστικός χρόνος για την αυτόνομη δράση των αντικειμένων του περιβάλλοντος [16].

### Κεφάλαιο 3 - Αλληλεπίδραση Χρήστη - Εικονικού Περιβάλλοντος

Στο κεφάλαιο αυτό αναλύεται ο τομέας της αλληλεπίδρασης μεταξύ χρήστη και εικονικού περιβάλλοντος (user interface). Θα μελετηθούν, αφενός, τρόποι παρουσίασης των τρισδιάστατων μοντέλων, που αποτελούν τον εικονικό κόσμο, στο χρήστη, αφετέρου δε, θα γίνει αναφορά στις μεθόδους εισόδου από το χρήστη. Η έξοδος από το σύστημα, το οποίο χρησιμοποιείται για την υλοποίηση του εικονικού περιβάλλοντος, αλλά και η είσοδος προς αυτό, είναι, προφανώς, απαραίτητες προϋποθέσεις για τη χρήση αλληλεπιδραστικών γραφικών (interactive graphics).

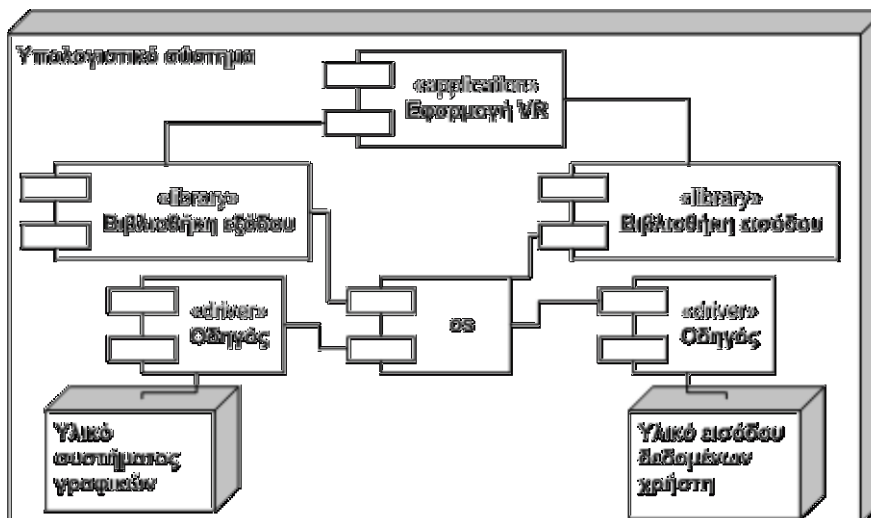
#### 3.1 Διακρίσεις Εικονικού Περιβάλλοντος

Στις εφαρμογές ΕΠ (VR) διακρίνονται δυο βασικά συστήματα: το σύστημα γραφικών και το σύστημα εισόδου χρήστη. Διακρίνει, επίσης, κανείς δυο χρήστες αυτών των συστημάτων: τον ανθρώπινο χρήστη και την εφαρμογή λογισμικού που υλοποιεί το εικονικό περιβάλλον. Οι σχέσεις μεταξύ όλων αυτών φαίνονται στο παρακάτω σχήμα. Το σύστημα γραφικών προβάλλει εικόνες στο χρήστη, ο οποίος, με τη σειρά του, εισάγει δεδομένα στο σύστημα εισόδου. Η προγραμματιστική εφαρμογή διαβάζει τα δεδομένα του χρήστη από το σύστημα εισόδου και δίνει εντολές στο σύστημα γραφικών, για να παράγει αυτό τα κατάλληλα γραφικά.



Εικόνα 19: Βασική δομή ενός συστήματος ΕΠ

Πρέπει να σημειωθεί ότι τα δυο συστήματα αποτελούνται από υλικό και λογισμικό. Η εφαρμογή προσπελαύνει το υλικό του συστήματος γραφικών μέσω μιας στοίβας προγραμματιστικών διασυνδέσεων (APIs τα οποία παρέχονται με τη μορφή βιβλιοθηκών), υπηρεσιών του λειτουργικού συστήματος, οδηγών των χρησιμοποιούμενων συσκευών. Αντίστοιχη στοίβα υπάρχει και στην περίπτωση της ανάγνωσης των δεδομένων εισόδου. Οι συσχετισμοί αυτοί αναπαρίστανται στο ακόλουθο σχήμα:



Εικόνα 20: Εφαρμογή ΕΠ και συστήματα εισόδου, εξόδου

### 3.2 Συστήματα Γραφικών

Το σύστημα γραφικών, τόσο το υλικό, όσο και το λογισμικό, πρέπει να επιλέγονται βάση ορισμένων χαρακτηριστικών, ώστε να ανταποκρίνονται στις απαιτήσεις της εφαρμογής. Η επιλογή αυτή που σχετίζεται με την παραγωγή των γραφικών, καθορίζονται κυρίως από το εάν η εφαρμογή κάνει χρήση εμβυθιστικής, ή όχι, εικονικής πραγματικότητας. Φυσικά σημαντικό ρόλο παίζει και η κατηγορία των χρηστών που πρόκειται να χρησιμοποιήσουν την εφαρμογή.

Το περιεχόμενο του εικονικού κόσμου που παρουσιάζεται στο χρήστη καθορίζεται από το είδος της εφαρμογής. Γενικά γίνεται προσπάθεια να είναι όσο το δυνατόν ρεαλιστικότερο. Παρόλα αυτά, υπάρχουν εφαρμογές που, σκοπίμως, χρησιμοποιούν μοντέλα με μη αληθοφανή όψη, πχ στυλ καρτούν για ιδιαίτερους, αισθητικούς λόγους. Η πολυπλοκότητα, όμως, των μοντέλων εξαρτάται από τις επεξεργαστικές δυνατότητες του υπολογιστικού συστήματος. Στη προκειμένη περίπτωση (των συνηθισμένων υπολογιστών), εξαρτάται από την ισχύ της κάρτας γραφικών.

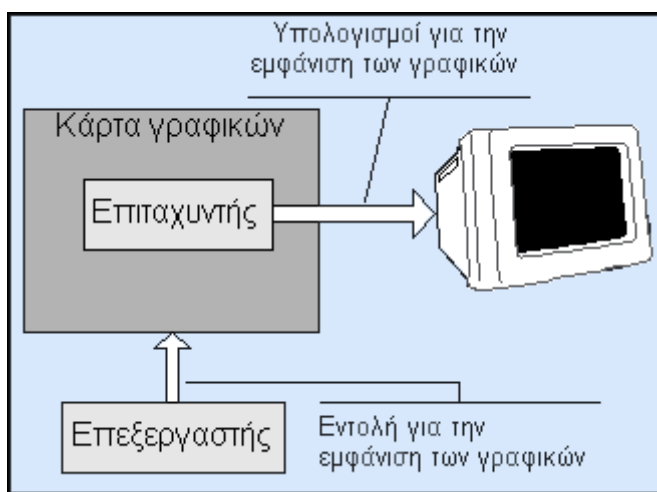
### 3.3 Λίγα Λόγια Για Τα Γραφικά Η/Υ

Η οθόνη του ηλεκτρονικού υπολογιστή είναι δισδιάστατη, συνεπώς, κάθε τι που ο υπολογιστής εμφανίζει, πρέπει να αναπαρίσταται σε δύο διαστάσεις. Για να μπορέσουμε να δουλέψουμε με 3D αντικείμενα θα πρέπει να μετατρέψουμε σε 2D εικόνες. Αυτό απαιτεί ειδική επεξεργασία και μεγάλη υπολογιστική ισχύ που πριν από λίγα χρόνια δεν ήταν διαθέσιμη. Αυτή την μετατροπή την επιτελούν οι νέες κάρτες γραφικών με δυνατότητα επεξεργασίας 3D, που εξαπλώθηκαν γρήγορα στην αγορά λόγω της ζήτησης. Η ζήτηση αφορούσε την ανάγκη για αυξημένη ρεαλιστικότητα, για πιο λεπτομερή γραφικά και για μεγαλύτερες ταχύτητες επεξεργασίας σε προγράμματα, όπως τα παιχνίδια δράσης, οι εξομοιωτές και τα σχεδιαστικά πακέτα, που επιτρέπουν στους χρήστες να κινούνται μέσα σε ένα εικονικό κόσμο [17].

Στις αρχές της δεκαετίας του '90, όταν τα γραφικά λειτουργικά συστήματα έγιναν δημοφιλή, οι κάρτες γραφικών δεν είχαν καθόλου λειτουργίες επιτάχυνσης. Η κάρτα γραφικών, άλλωστε, είναι μόνο ένα μέρος του συνόλου που καθορίζει τι θα δούμε στην οθόνη. Είναι, κατά κάποιον τρόπο, ο "μεσολαβητής" μεταξύ του επεξεργαστή και της οθόνης. Η οθόνη είναι αυτή που, στην πράξη, παρέχει αυτό που βλέπουμε, ενώ ο επεξεργαστής είναι αυτός που υπολογίζει και αποφασίζει τι πρόκειται να δούμε. Μια συμβατική κάρτα γραφικών κάνει τη δουλειά της μετατροπής αυτών που παράγει ο επεξεργαστής σε μορφή που μπορεί η οθόνη να τα εμφανίσει. Μάλιστα, παλαιότερα, το λειτουργικό Σύστημα ανάγκαζε τον επεξεργαστή του

συστήματος να κάνει όλη τη δουλειά της εμφάνισης των γραφικών στην οθόνη, γεγονός που καθυστερούσε δραματικά την απόδοση του συστήματος [17].

Για να λυθεί αυτό το πρόβλημα και να ελαττωθεί το υπολογιστικό βάρος που επωμιζόταν ο επεξεργαστής του συστήματος, σχεδιάστηκαν επιταχυντές γραφικών που έκαναν τη δουλειά της μετατροπής μιας 3D εικόνας σε 2D με τη χρήση ειδικού hardware. Πρόκειται για κάρτες γραφικών, στις οποίες έχει προστεθεί η δυνατότητα να κάνουν το μεγαλύτερο μέρος των υπολογισμών για την εμφάνιση της εξόδου, δουλειά που προηγουμένως γινόταν από τον επεξεργαστή. Με έναν επιταχυντή, όταν το σύστημα χρειάζεται να εμφανίσει ένα παράθυρο στην οθόνη, δεν υπολογίζει ποιά ακριβώς εικονοστοιχεία πρέπει να φωτιστούν και με τι χρώμα, απλά στέλνει μια εντολή στην κάρτα γραφικών για να σχεδιάσει ένα παράθυρο σε κάποια περιοχή και η κάρτα γραφικών το υλοποιεί. Ο επιταχυντής μπορεί να είναι προσαρμοσμένος και εξειδικευμένος σε αυτή τη δουλειά και συνεπώς, μπορεί να είναι πιο αποδοτικός σε σχέση με τον επεξεργαστή. Από την άλλη, με αυτόν τον τρόπο, ο επεξεργαστής μπορεί να συνεχίσει να κάνει πιο χρήσιμους υπολογισμούς [17].



**Εικόνα 21: Λειτουργία κάρτας γραφικών**

Την σημερινή εποχή είναι αναγκαίο να υπάρχουν 3D κάρτες γραφικών για να δημιουργούν τα 3D γραφικά, αλλά το μέγεθος των υπολογισμών που απαιτούνται για να μετατραπούν οι 3D εικόνες σε 2D με ρεαλιστικό τρόπο γίνεται με ειδικό hardware. Χρησιμοποιώντας τους 3D επιταχυντές, επιτρέπουμε στα προγράμματα να εμφανίζουν 3D εικονικούς κόσμους με ένα επίπεδο λεπτομέρειας και χρώματος που με τις τυπικές 2D κάρτες γραφικών ήταν αδύνατο, ενώ παράλληλα απελευθερώνουμε τον επεξεργαστή του συστήματος από την εκτέλεση μιας επιπλέον πολύπλοκης και χρονοβόρας διεργασίας [17].

Όσον αφορά τις ίδιες τις τρισδιάστατες εικόνες, οφείλουμε να πούμε πως είναι πολύ πιο πολύπλοκες από τις δισδιάστατες, λόγω των πολύ μεγαλύτερων ποσών πληροφορίας που χρησιμοποιούνται για να δημιουργηθεί ένας ρεαλιστικός 3D κόσμος. Επιπρόσθετα, πολλές μαθηματικές λειτουργίες πρέπει να χρησιμοποιηθούν για να μετατρέψουν τον 3D κόσμο σε εικόνα που μπορεί να εμφανιστεί στην οθόνη του υπολογιστή. Όταν κοιτάμε τον πραγματικό κόσμο, τα μάτια και ο εγκέφαλος μας εκτελούν αυτή την ενέργεια αυτόματα και οι περισσότερες από τις λειτουργίες που μας επιτρέπουν να αντιλαμβανόμαστε έναν 3D κόσμο συμβαίνουν τόσο γρήγορα που δεν αντιλαμβανόμαστε ότι γίνονται. Το πώς αντιλαμβανόμαστε τον κόσμο, είναι ένα αποτέλεσμα της σύνθετης αλληλεπίδρασης κάποιων οπτικών εφέ, όπως το επίπεδο του φωτός, η σκίαση, η σχετική κίνηση και ο τρόπος που γνωρίζουμε ότι λειτουργεί ο κόσμος. Δουλειά της μηχανής γραφικών είναι να εξομοιώνει αυτή τη φυσική λειτουργία όσο είναι δυνατό, έτσι ώστε να βλέπουμε στην οθόνη να μας φαίνεται ρεαλιστικό [17].

Οι 3D εικόνες επεξεργάζονται στον υπολογιστή χρησιμοποιώντας τη μοντελοποίηση (modeling). Τα 3D μοντέλα αντιπροσωπεύουν ένα 3D αντικείμενο χρησιμοποιώντας μια

συλλογή σημείων και άλλων πληροφοριών στον τρισδιάστατο χώρο, τα οποία συνδέονται μεταξύ τους με διάφορες γεωμετρικές οντότητες όπως τρίγωνα, ευθύγραμμα τμήματα, καμπύλες.

Τα μοντέλα μπορούν να δημιουργηθούν είτε χειροκίνητα είτε με αλγοριθμικές διαδικασίες (procedural modeling) ή μέσω σάρωσης (model scanning) [18].

Δυο είδη 3D μοντέλων είναι τα παρακάτω:

- **Στερεά (Solid):** Χρησιμοποιούνται κυρίως για μη γραφικές προσομοιώσεις όπως για παράδειγμα ιατρικές ή μηχανικές. Τα μοντέλα αυτά καθορίζουν τον όγκο του αντικειμένου που αντιπροσωπεύουν.
- **Όρια (Shell/Boundary):** Είναι ευκολότερα στη χρήση από τα στερεά μοντέλα. Καθορίζουν την επιφάνεια, π.χ. το όριο του αντικειμένου, και όχι τον όγκο του.

Οι διαδικασίες μοντελοποίησης (modeling) είναι οι παρακάτω [18]:

- **Polygonal Modeling:** Σημεία, κορυφές στον τρισδιάστατο χώρο συνδέονται με γραμμικά τμήματα σχηματίζοντας πολύγωνα. Η φιλοσοφία των περισσότερων 3D μοντέλων σήμερα είναι χτισμένη πάνω σε πολυγωνικά μοντέλα, επειδή είναι ευέλικτα και επειδή οι υπολογιστές μπορούν να τα επεξεργαστούν σε πολύ μικρό χρόνο. Επειδή βέβαια, τα πολύγωνα είναι επίπεδες επιφάνειες, οι σύνθετες κυρτές επιφάνειες μοντελοποιούνται μόνο κατά προσέγγιση με τη χρήση πολλών πολυγώνων.
- **NURBS Modeling:** Επιφάνειες NURBS ορίζονται από spline καμπύλες οι οποίες επηρεάζονται από σταθμισμένα σημεία ελέγχου (weighted control points). Η αύξηση του βάρους για ένα σημείο θα τραβήξει την καμπύλη πιο κοντά στο σημείο αυτό. Τα NURBS είναι πραγματικά λείες επιφάνειες και όχι απλές προσεγγίσεις που χρησιμοποιούν μικρές επίπεδες επιφάνειες και έτσι, είναι ιδιαίτερα κατάλληλες για οργανικές μοντελοποιήσεις.
- **Primitives Modeling:** Αυτή η διαδικασία θεωρεί τα πρωτογενή γεωμετρικά σχήματα (όπως μπάλες, κύλινδροι, κώνοι, κύβοι) ως δομικά στοιχεία για πιο πολύπλοκα μοντέλα. Εδώ οι μορφές ορίζονται με μαθηματικό τρόπο, με αποτέλεσμα να είναι απόλυτα ακριβείς. Επίσης, η γλώσσα ορισμού τους μπορεί να είναι πολύ απλούστερη. Γενικότερα, αποτελεί μια εύκολη και γρήγορη κατασκευή μοντέλων κατάλληλη για τεχνικές εφαρμογές.
- **Splines & Patches Modeling:** Εξαρτώνται από καμπυλωτές γραμμές για να καθορίσουν την ορατή επιφάνεια. Σε ότι αφορά στην ευελιξία και στην ευκολία χρήσης βρίσκονται κάπου μεταξύ καμπυλών NURBS και polygonal.
- **Απόδοση Σχεδιοκίνησης (animation):** Διακρίνονται σε τρεις βασικές μεθόδους:
  - Η μέθοδος key frames (σημαντικών καρέ): Χρησιμοποιείται στα περισσότερα προγράμματα κατασκευής 3D. Τα μοντέλα τοποθετούνται σε σημαντικά χρονικά σημεία σε συγκεκριμένες θέσεις του κόσμου και το πρόγραμμα αναλαμβάνει να συμπληρώσει τα ενδιάμεσα καρέ βάσει της τροχιάς της κίνησης που έχει οριστεί.
  - Η μέθοδος παραμετρικών key frames: Έχει την ίδια λογική με την προηγούμενη μέθοδο μόνο που εδώ η κάθε οντότητα (αντικείμενο, κάμερα, φως) χαρακτηρίζεται από παραμέτρους.
  - Η μέθοδος του διαδικαστικού (procedural) animation: Είναι μια αλγοριθμική μέθοδος στην οποία χρησιμοποιούνται χωρικές και χρονικές μετατροπές (περιστροφή, μετακίνηση), οι οποίες καθορίζονται από παραμέτρους (π.χ. γωνία περιστροφής) οι οποίες μπορούν να αλλάξουν κατά τη διάρκεια του animation.

Γενικότερα έχουν αναπτυχθεί διάφορες *τεχνικές τρισδιάστατης σχεδιοκίνησης* όπως [18]:

- ο η κινηματική (kinimatics) η οποία αφορά τις ιδιότητες των αντικειμένων, όπως η θέση, η ταχύτητα και η επιτάχυνση. Σε περιπτώσεις όπου το αντικείμενο είναι τεμαχισμένο σε περισσότερα κομμάτια τότε τα κομμάτια αυτά συνδέονται μεταξύ τους δημιουργώντας μια δενδρική ιεραρχία.
- ο η δυναμική (dynamic), η οποία είναι αυτή που θα δώσει στο αντικείμενο τις φυσικές του ιδιότητες, λαμβάνοντας υπόψη τους νόμους της φυσικής και προσθέτοντας στην κίνηση του αντικειμένου χαρακτηριστικά ρεαλιστικότητας. Εδώ χρησιμοποιούνται στοιχεία όπως το υλικό, το βάρος, το μέγεθος, η πυκνότητα.

Ουσιαστικά, κάθε 3D αντικείμενο συντίθεται από εκατοντάδες ή και χιλιάδες μικρά τρίγωνα (ή άλλα πολύγωνα) που περιγράφουν τη δομή του. Όταν το πρόγραμμα θέλει να μετακινήσει ένα αντικείμενο, μπορούμε να πούμε, γενικά και απλουστευμένα, πως αυτό που κάνει είναι ότι αλλάζει τις γωνίες αυτών των τριγώνων για να δημιουργήσει κίνηση. Η μεγαλύτερη υπολογιστική δουλειά σχετίζεται με τη μετατροπή αυτών των καμπύλων τριγώνων σε μια συμπαγή επιφάνεια. Φυσικά, τα πραγματικά αντικείμενα δεν είναι δημιουργημένα με αυτό τον τρόπο, αλλά αυτή η τεχνική είναι αναγκαία για να πετύχουμε προσομοίωση της κίνησης. Επίσης, στον πραγματικό κόσμο, τα αντικείμενα δεν είναι απομονωμένα, αλλά αλληλεπιδρούν μεταξύ τους. Καλύπτει το ένα το άλλο, προκαλούν σκιές, αντανακλούν το φως και εμφανίζονται πιο θαμπά όταν είναι σε απόσταση. Υπάρχουν πολύ σύνθετες μαθηματικές συναρτήσεις που χρησιμοποιούνται για το κατά πόσο ένα αντικείμενο θα είναι ορατό και τι χρώμα πρέπει να έχει. Αν παίζουμε ένα 3D παιχνίδι και θέλουμε πιστή απεικόνιση, αυτοί οι υπολογισμοί πρέπει να γίνουν είκοσι και πλέον φορές το δευτερόλεπτο. Γι' αυτό χρησιμοποιούνται οι 3D επιταχυντές, οι οποίοι είναι προσαρμοσμένοι έτσι ώστε να βελτιώνουν την απόδοση αυτών των υπολογισμών [18].

Κάθε φορά που η εικόνα υπολογίζεται ξανά (για παράδειγμα, εξαιτίας μιας κίνησης σε ένα ηλεκτρονικό παιχνίδι), είναι αναγκαίο να υπολογιστεί ξανά το χρώμα και η ένταση κάθε εικονοστοιχείου πάνω στη 2D οθόνη. Υπάρχουν διάφοροι τύποι υπολογισμών που χρησιμοποιούνται στην 3D επεξεργασία. Κάποιες κάρτες υποστηρίζουν περισσότερους υπολογισμούς σε σχέση με άλλες και κάποιες είναι πιο αποδοτικές σε συγκεκριμένους υπολογισμούς σε σχέση με άλλες. Μερικές από τις πιο συνηθισμένες 3D λειτουργίες είναι οι παρακάτω [18]:

- **Φωτοσκίαση:** Πρόκειται για έναν αλγόριθμο που χρησιμοποιείται για να δώσει ρεαλιστική σκίαση σε 3D επιφάνειες. Αυτό προσθέτει βάθος στο αντικείμενο που εμφανίζεται και βοηθά στο να προσδιορίζεται καλύτερα το σχήμα του.
- **Αποκοπή:** Αυτή η λειτουργία καθορίζει ποιο μέρος του αντικειμένου εμφανίζεται στην οθόνη και αποκόπτει τα μέρη που ο χρήστης δεν μπορεί να δει. Εξοικονομεί χρόνο, καθώς τα μέρη που δεν φαίνονται απλά αγνοούνται.
- **Φωτισμός:** Τα αντικείμενα στον πραγματικό κόσμο οφείλουν τη διαμόρφωση της εμφάνισής τους στις πηγές του φωτός του χώρου. Ο φωτισμός, λοιπόν, προκαλεί την αντανάκλαση, τη σκίαση και άλλα εφέ που προστίθενται στο αντικείμενο και εμφανίζονται σε σχέση με τη θέση του αντικειμένου και την πηγή του φωτός στο χώρο.
- **Διαφάνεια:** Κάποια αντικείμενα στον πραγματικό κόσμο είναι διάφανα ή ημιδιάφανα. Ειδικοί υπολογισμοί απαιτούνται, για να καθορίσουν τη διαφάνεια ενός αντικειμένου, για παράδειγμα ποιά αντικείμενα θα είναι ορατά πίσω από ένα τζάμι.
- **Διαμόρφωση υφής:** Σε ρεαλιστικά αντικείμενα, είναι αναγκαίο να επικαλυφθούν κάποιες εικόνες πάνω τους, έτσι ώστε τα αντικείμενα αυτά να αποκτήσουν μια υφή. Η διαμόρφωση υφής επιτρέπει στα αντικείμενα να εμφανίζονται σαν να έχουν υπόσταση. Στην πραγματικότητα, υπάρχουν αρκετοί διαφορετικοί τύποι για διαμόρφωση υφής που χρησιμοποιούνται από το λογισμικό και το hardware.



- **Συσκότιση:** Πρόκειται για εφέ που χρησιμοποιείται για την αναπαράσταση εξωτερικών χώρων, θαμπώνοντας τα αντικείμενα που βρίσκονται σε απόσταση. Εξυπηρετεί δυο σκοπούς: πρώτον, βοηθάει να γίνει πιο ρεαλιστική η απεικόνιση και δεύτερον, βοηθάει τη 3D επεξεργασία να ολοκληρωθεί πιο γρήγορα, καθώς τα αντικείμενα που βρίσκονται σε απόσταση, μπορούν να υπολογιστούν πολύ γρήγορα, αφού δε χρειάζονται πολλές λεπτομέρειες.
- **Φιλτράρισμα:** Υπάρχουν διάφοροι τύποι φιλτραρίσματος που μπορούν να εφαρμοστούν σε μια εικόνα. Χρησιμοποιούνται για να "καθαρίσουν" την εικόνα και να μαλακώσουν την υφή και τα σχήματα.

### 3.4 Η Λειτουργία Buffering

Δεν πρόκειται πραγματικά για μια 3D λειτουργία, όπως οι άλλες που αναφέρθηκαν παραπάνω, καθώς δεν είναι κάτι που έχει να κάνει με τα δεδομένα. Ωστόσο, οι εξελιγμένες κάρτες γραφικών περιλαμβάνουν καταχωρητές μνήμης (memory buffers), που χρησιμοποιούνται για διάφορες λειτουργίες κατά τη διάρκεια των πολύπλοκων υπολογισμών. Όσους περισσότερους καταχωρητές έχει μια κάρτα, τόσο μεγαλύτερη ευελιξία έχει στο να κάνει κάποιες συγκεκριμένες λειτουργίες. Αυτό οφείλεται στο γεγονός ότι οι 3D κάρτες χρειάζονται συνήθως περισσότερη μνήμη από όση θα χρειάζονταν για να κρατούν απλώς την εικόνα. Τα καινούρια AGP (Accelerated Graphics Port) συστήματα μπορούν να χρησιμοποιήσουν τη μνήμη του συστήματος για αυτό το σκοπό [18].

### 3.5 Η Διαδικασία Του Rendering

Το rendering είναι μια διαδικασία των γραφικών του υπολογιστή που κάνει αυτόματα τη μετατροπή 3D μοντέλων σε δισδιάστατες εικόνες, με ή χωρίς τρισδιάστατα φωτορεαλιστικά εφέ. Στην ουσία, πρόκειται για το τελευταίο στάδιο δημιουργίας μιας δισδιάστατης εικόνας ή ενός animation από μια ψηφιακή σκηνή που έχει στηθεί. Μπορεί να παραλληλιστεί στον φυσικό κόσμο με το τράβηγμα μιας φωτογραφίας ή με την κινηματογράφηση μιας σκηνής, αφού πρώτα έχουν στηθεί όλα τα σκηνικά. Το rendering μπορεί να χρειαστεί κλάσματα του δευτερολέπτου ή μέρες ολόκληρες για να αποδώσει μια μόνο εικόνα/πλαίσιο. Σε γενικές γραμμές, οι διαφορετικές τεχνικές που έχουν αναπτυχθεί εφαρμόζονται, ανάλογα τις ανάγκες, είτε για φωτο-ρεαλιστική απόδοση είτε για απόδοση σε πραγματικό χρόνο (real-time rendering), όπως συμβαίνει στην περίπτωση των εικονικών περιβαλλόντων [19].

Στο real-time rendering ο στόχος είναι να μεταδοθεί όση περισσότερη πληροφορία δύναται να επεξεργαστεί το ανθρώπινο μάτι σε ένα κλάσμα του δευτερολέπτου (ή αλλιώς σε ένα πλαίσιο). Ο πρωτεύον στόχος είναι να επιτευχθεί ο πιο υψηλός βαθμός φωτορεαλισμού στην ελάχιστη επιτρεπτή ταχύτητα rendering, που συνήθως αντιστοιχεί σε 24 πλαίσια/sec, καθώς είναι ο ελάχιστος ρυθμός που μπορεί να ξεγελάσει το ανθρώπινο μάτι, δημιουργώντας την ψευδαίσθηση της κίνησης [20].

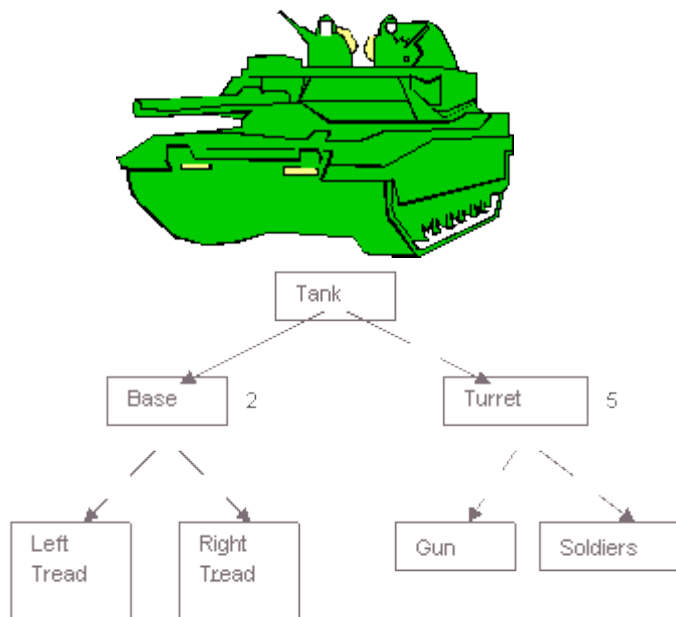
Το real-time rendering αφορά πολυγωνικά αντικείμενα και ενισχύεται από τη Μονάδα Επεξεργασίας Γραφικών/GPU (Graphical Process Unit) του υπολογιστή. Η αύξηση της επεξεργαστικής ισχύος των υπολογιστών τα τελευταία χρόνια έχει συνεισφέρει στην αύξηση της ταχύτητας του rendering, ενώ παράλληλα έχει μειωθεί το αντίστοιχο οικονομικό κόστος [19].

### 3.6 Γράφημα Σκηνής

Ένα γράφημα σκηνής είναι ένας τρόπος να ταξινομείς ιεραρχικά τα δεδομένα μιας σκηνής, δηλαδή είναι μια δομή δεδομένων με κόμβους-γονείς και κόμβους-παιδιά, όπου η σχέση γονέα-παιδιού είναι θεμελιώδης και η σειρά των κόμβων είναι σημαντική, καθώς τα παιδιά κληρονομούν ιδιότητες των γονέων. Το γράφημα σκηνής χρησιμοποιείται συχνά σε σύγχρονα βιντεοπαιχνίδια και σε εφαρμογές επεξεργασίας γραφικών που χρησιμοποιούν διανύσματα

(vector-based graphics), όπως είναι οι δύο γλώσσες γραφικών αναπαράστασεων που θα χρησιμοποιήσουμε, η VRML97 και η XJ3D [21][22].

Το γράφημα σκηνής είναι μια δομή που ορίζει τη λογική και συνήθως και την χωρική αναπαράσταση μιας σκηνής γραφικών. Αν και ο ορισμός του γραφήματος σκηνής ποικίλει, λόγω του διαφορετικού τρόπου εφαρμογής της από τους προγραμματιστές σε διαφορετικές εφαρμογές, εντούτοις θα περιγράψαμε το γράφημα σκηνής ως μια συλλογή κόμβων σε μια δομή γραφήματος ή δέντρου [21][22].



**Εικόνα 22: Παράδειγμα γραφήματος σκηνής**

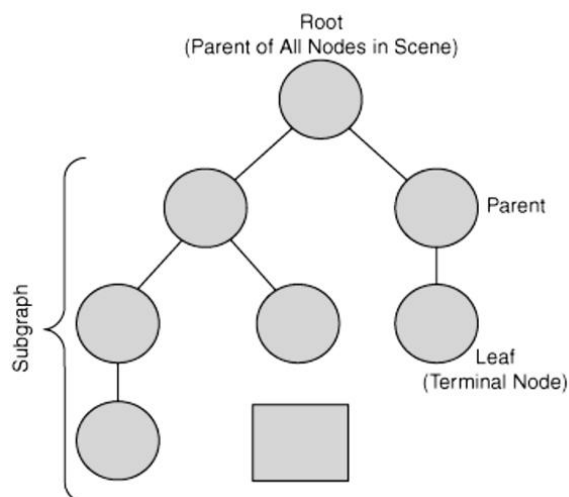
Ένα δέντρο μπορεί να έχει πολλά παιδιά, αλλά συνήθως έχει έναν μοναδικό γονέα, με τις ιδιότητές του να εφαρμόζονται σε όλους τους κόμβους-παιδιά, οπότε, το αποτέλεσμα μιας λειτουργίας που εφαρμόζεται σε μια ομάδα αυτόματα μεταδίδεται σε όλα τα μέλη της ομάδας. Για παράδειγμα, μια χαρακτηριστική ιδιότητα είναι η ικανότητα ομαδοποίησης σχετικών σχημάτων/αντικειμένων για τη δημιουργία ενός ενιαίου, συμπαγούς αντικειμένου που μπορεί να μετατοπιστεί, να μετασχηματιστεί και να επιλεγεί σαν να επρόκειτο για ένα ξεχωριστό, αυτόνομο αντικείμενο.

Το γράφημα σκηνής αποτελεί μια χρήσιμη αφαιρετική μέθοδο, τόσο για τα δεδομένα της σκηνής όσο και για τη συμπεριφορά της, καθώς παλαιότερα, αυτά τα δύο ορίζονταν διαδικαστικά. Ο κώδικας που όριζε τη σκηνή ήταν διεσπαρμένος μαζί με τον κώδικα που όριζε τη λειτουργικότητα της σκηνής. Κατά αυτόν τον τρόπο, αλλαγές στη σκηνή ή στη συμπεριφορά συνεπάγονταν αλλαγές στον κώδικα και το αποτέλεσμα ήταν ένας πολύπλοκος και ανελαστικός κώδικας, δύσκολος στη δημιουργία, την τροποποίηση και τη συντήρηση. Χωρίζοντας τη σκηνή από τις λειτουργίες που εφαρμόζονται σε αυτήν, το προγραμματιστικό μοντέλο του γραφήματος σκηνής εγκαθιδρύει ένα καθαρό όριο ανάμεσα στην αναπαράσταση της σκηνής και την απόδοση της στην κάρτα γραφικών (rendering). Συνεπώς, η σκηνή μπορεί να συντεθεί και να συντηρηθεί ανεξάρτητα από τις μεθόδους που λειτουργούν επ' αυτής, δίνοντας έμφαση στο οπτικό περιεχόμενο της σκηνής και στην όποια συμπεριφορά και συσχέτιση υπάρχει μεταξύ των αντικειμένων σε αυτήν και όχι στο πώς αυτό το περιεχόμενο υφίσταται επεξεργασία σε χαμηλότερο επίπεδο. Η υποβόσκουσα υλοποίηση και οι λεπτομέρειες του rendering υπεξαιρούνται από το προγραμματιστικό μοντέλο του γραφήματος σκηνής, πχ. στην VRML είναι το plug-in του browser υπεύθυνο για αυτήν την χαμηλού επιπέδου αναπαράσταση [21][22].

Ένα γράφημα σκηνής αποτελείται από κόμβους (που αναπαριστούν τα αντικείμενα στη σκηνή) και ακμές (που συνδέουν τους κόμβους και ορίζουν συσχετίσεις μεταξύ τους).

Συνδυαστικά, οι κόμβοι και οι ακμές παράγουν ένα γράφημα που οργανώνει τα αντικείμενα σε μια ιεραρχική δομή, βάσει της χωρικής τους θέσης στη σκηνή. Με την εξαίρεση του κόμβου-ρίζα, που ορίζει το σημείο έναρξης του γραφήματος σκηνής, κάθε κόμβος έχει έναν γονέα. Οι κόμβοι που εμπεριέχουν άλλους κόμβους είναι κόμβοι-γονείς, ενώ οι κόμβοι που εμπεριέχονται είναι κόμβοι-παιδιά. Κόμβοι που περιέχουν παιδιά είναι κόμβοι ομαδοποίησης, ενώ οι τελευταίοι κόμβοι στην ιεραρχία ορίζονται ως κόμβοι-φύλλα. Λειτουργίες επί της σκηνής μπορούν να εφαρμοστούν σε όλους τους κόμβους ή μπορούν να περιοριστούν σε ένα συγκεκριμένο υποσύνολο του γραφήματος, συνεπώς μια σκηνή μπορεί να συντίθεται από ξεχωριστούς κόμβους ή από ολόκληρα υποσύνολα που μπορούν να ενσωματωθούν ή να αποσπαστούν από τη σκηνή, σύμφωνα με τις εκάστοτε ανάγκες.

Ιδωμένο οπτικά, ένα γράφημα σκηνής θυμίζει μια δενδροειδή δομή. Η δομή δεδομένων DAG (Directed Acyclic Graph – Κατευθυνόμενος Κυκλικός Γράφος) χρησιμοποιείται συχνά, επειδή υποστηρίζει διαμοίρασμα των κόμβων (nodesharing) σε υψηλό επίπεδο στον γράφο. Οι κόμβοι στη δομή DAG μπορούν να έχουν πάνω από ένα γονέα, εις βάρος, ωστόσο, επιπλέον πολυπλοκότητας στον κώδικα. Στο DAG όλοι οι κόμβοι πρέπει να έχουν μια ορισμένη κατεύθυνση σχέσης γονέα – παιδιού στην οποία δεν επιτρέπονται κύκλοι/επαναλήψεις [21][22].

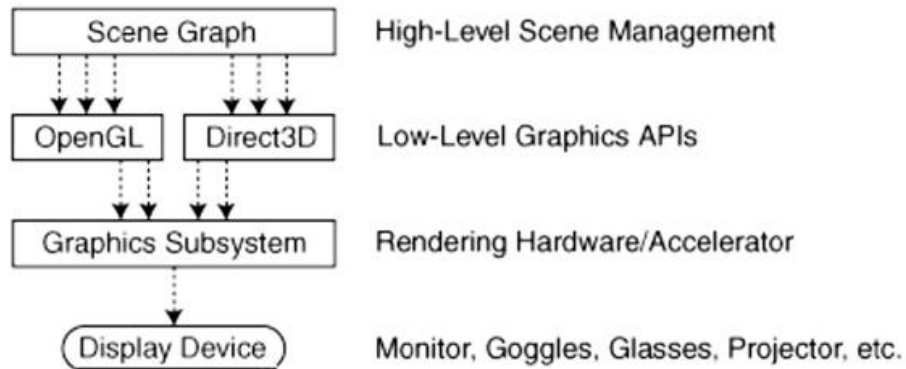


**Εικόνα 23: Δομή Γραφήματος Σκηνής**

Τα γραφήματα σκηνής που χρησιμοποιούνται για τριδιάστατο περιεχόμενο, συνήθως υποστηρίζουν κόμβους που αναπαριστούν πρωτογενή τρισδιάστατα γεωμετρικά σχήματα (προδηλωμένα τετράγωνα, κώνους, σφαίρες), πολύπλοκα πολυγωνικά σχήματα, φώτα, υλικό, ήχο. Επίσης, υπάρχουν κόμβοι που μπορεί να υποστηρίξουν οπτικοακουστικό περιεχόμενο, συγχρονισμό, ειδικά εφέ και άλλες λειτουργικότητες για τη σύνθεση πολυμέσων (multimedia).

Το γράφημα σκηνής υποστηρίζει διάφορες πράξεις, μέσω διάσχισης της δομής του γράφου με κατεύθυνση από τη ρίζα προς τα φύλλα. Η διάσχιση του γράφου είναι απαραίτητη για την εκτέλεση ενός πλήθους λειτουργιών που αφορούν ενέργειες του rendering και σχετίζονται με μετασχηματισμούς, αποκοπές, φωτισμό και άλλες λειτουργίες, πιο διαδραστικές, όπως η ανίχνευση συγκρούσεων και η επιλογή αντικειμένων. Οι κόμβοι που επηρεάζονται από μια δοσμένη λειτουργία διατρέχονται κατά τη διάρκεια μιας διάσχισης. Φτάνοντας στον κόμβο, η εσωτερική του κατάσταση μπορεί να μεταβληθεί (εφόσον υποστηρίζεται κάτι τέτοιο από την αντίστοιχη γλώσσα), ώστε να ανακλά την κατάσταση της συγκεκριμένης λειτουργίας εκείνη τη στιγμή. Οι διάσχισεις κατά το rendering συμβαίνουν διαρκώς με τα διαδραστικά γραφικά, επειδή η κατάσταση των πραγμάτων αλλάζει όσο συχνά αλλάζει και η οπτική γωνία του χρήστη, οδηγώντας σε συνεχή ερωτήματα (queries) στο γράφημα σκηνής και ανανεώσεις σε μια διαρκώς εναλλασσόμενη προοπτική. Τέλος, ένα γράφημα σκηνής μπορεί να θεωρηθεί ως μια

υψηλού επιπέδου περιγραφή των API γραφικών χαμηλού επιπέδου, όπως η OpenGL και η DirectX [21][22].



Εικόνα 24: Ροή υλοποίησης γραφήματος σκηνής

## Κεφάλαιο 4 - Εικονικοί - Ευφυής Πράκτορες

Όπως αναφέρθηκε, στα περιβάλλοντα αυτά λειτουργούν οντότητες, γνωστές ως εικονικοί πράκτορες (virtual agents), που μπορούν να επιδράσουν με το περιβάλλον και να προκαλέσουν κάποιου είδους μεταβολή σε αυτό. Υπάρχουν πολλοί ορισμοί για το τι είναι πράκτορας και αυτό οφείλεται στο ότι υπάρχουν πολλές διαφορετικές προσεγγίσεις. Οι αυτόνομες ή ημιαυτόνομες προγραμματιστικές οντότητες έχει επικρατήσει να αποκαλούνται πράκτορες, και έχουν δημιουργήσει έναν θρύλο γύρω τους που παραπέμπει σε έξυπνα ρομπότ που παρέχουν κάθε είδους υπηρεσίες στον άνθρωπο ή και σε προγράμματα ιών που εισέρχονται στο σύστημα μας και καταστρέφουν τα δεδομένα μας.

### 4.1 Τι Είναι Πράκτορας

Πράκτορας (agent) είναι οτιδήποτε μπορεί να θεωρηθεί ότι αντιλαμβάνεται το περιβάλλον (environment) του μέσω αισθητήρων (sensors) και αλληλεπιδρά με αυτό μέσω μηχανισμών δράσης (effectors – actuators) [23].

Ο όρος “πράκτορας” παρατηρούμε ότι δεν είναι μία σαφώς καθορισμένη έννοια αφού έχουν προταθεί από διάφορους ερευνητές πολλοί διαφορετικοί ορισμοί. Έτσι, αυτό που θα κάνουμε στη συνέχεια είναι όχι να δώσουμε κάποιον αυστηρό ορισμό, αλλά να περιγράψουμε τις κυριότερες ιδιότητες ενός “πράκτορα” έτσι ώστε να δοθεί μία γενική εικόνα της έννοιας αυτής [23].

### 4.2 Εικονικοί Πράκτορες Σήμερα

Τα τελευταία χρόνια έχουν λάβει χώρα σημαντικές προσπάθειες στον τομέα της ενίσχυσης της αυτονομίας των εικονικών περιβαλλόντων με την ανάπτυξη ενός νέου πεδίου έρευνας στο χώρο αυτό, των εικονικών πρακτόρων [23].

Εικονικοί πράκτορες (virtual agents) ονομάζονται οι αυτόνομες οντότητες σε ένα εικονικό περιβάλλον, οι οποίες έχουν συνήθως τη μορφή συνθετικών χαρακτήρων, και αλληλεπιδρούν με το περιβάλλον μέσω αισθητήρων (sensors) και επιδραστών (effectors) λαμβάνοντας αποφάσεις με βάση κάποιο μοντέλο συμπεριφοράς. Οι χαρακτήρες αυτοί μπορούν να είναι άνθρωποι, ζώα, ή ακόμα και φανταστικά πλάσματα, ενώ σε κάποιες περιπτώσεις εικονικοί πράκτορες μπορούν να είναι και μηχανήματα με πολύπλοκη συμπεριφορά, όπως αυτοκίνητα, ρομπότ. Οι εικονικοί πράκτορες παρουσιάζουν μεγάλη χρησιμότητα στα περισσότερα από τα πεδία εφαρμογής των εικονικών περιβαλλόντων, καθώς η παρουσία αυτόνομων χαρακτήρων σε τρισδιάστατους κόσμους ενισχύει την αληθοφάνειά τους αλλά και τις δυνατότητες αλληλεπίδρασης με τον χρήστη.

Αν και η υπάρχουσα έρευνα στους εικονικούς πράκτορες φαίνεται να βρίσκεται σε πρωταρχικά στάδια, καθώς υπάρχουν πολλά προβλήματα που δεν έχουν ακόμα επιλυθεί, μια πρώτη προσπάθεια ταξινόμησης των υπάρχοντων συστημάτων καταλήγει σε ένα φάσμα πρακτόρων [23].

Στη μία άκρη του φάσματος τοποθετούνται οι φυσικοί πράκτορες (physical agents), δηλαδή αυτοί στους οποίους η έμφαση έχει δοθεί στην αληθοφανή φυσική συμπεριφορά. Στην άλλη άκρη του φάσματος αυτού τοποθετούνται οι γνωσιακοί πράκτορες (cognitive agents), δηλαδή αυτοί στους οποίους η έμφαση έχει δοθεί στην ανθρώπινη γνωσιακή συμπεριφορά, καθώς και στη γνωσιακή αλληλεπίδραση με τους χρήστες του συστήματος [23].

Τα προβλήματα της έλλειψης εργαλείων ανάπτυξης στο χώρο των εικονικών περιβαλλόντων αντικατοπτρίζονται, όπως είναι φυσικό, και στην εξειδικευμένη περιοχή των εικονικών πρακτόρων. Έτσι, ενώ τα τελευταία χρόνια, κυρίως λόγω της εκτενούς χρήσης των τρισδιάστατων γραφικών στα παιχνίδια με υπολογιστή, έχουν κατασκευαστεί κάποιες μηχανές τρισδιάστατων γραφικών με δυνατότητα ενσωμάτωσης συνθετικών χαρακτήρων, είναι ευθύνη του σχεδιαστή να προγραμματίσει τη συμπεριφορά τους και να τους κάνει να συμπεριφέρονται με αληθοφανή τρόπο. Η μηχανή από μόνη της δεν εγγυάται ότι οι συνθετικοί χαρακτήρες συμπεριφέρονται όπως θα έπρεπε και δεν παρέχει κάποιο μηχανισμό, για να προγραμματιστεί

και να ρυθμιστεί εύκολα η συμπεριφορά τους. Το γεγονός αυτό δε βοηθάει ιδιαίτερα στην επαναχρησιμοποίηση των πρακτόρων, καθώς θα χρειαστεί αρκετός επιπλέον προγραμματισμός, για να συνδεθούν τμήματα ενός πράκτορα σε κάποια άλλη εφαρμογή [23].

Επιπροσθέτως, υπάρχουν ορισμένα χαρακτηριστικά και διαδικασίες των εικονικών πρακτόρων, όπως η χρήση κάποιας εσωτερικής μνήμης, η αντίληψη και η δυνατότητα πραγματοποίησης σύνθετων ενεργειών, που φαίνεται να είναι απαραίτητες σε οποιοδήποτε περιβάλλον απαιτεί σύνθετη συμπεριφορά και αληθοφάνεια. Τα παραπάνω χαρακτηριστικά θα έπρεπε λοιπόν να αποτελούν μέρος της λειτουργικότητας του πράκτορα και όχι να προστίθενται κατά την διάρκεια ανάπτυξης της εφαρμογής [23].

### **4.3 Διαχωρισμός Πρακτόρων**

Οι εικονικοί πράκτορες χωρίζονται σε φυσικούς και γνωσιακούς. Στους πρώτους δίνεται έμφαση στην κίνηση και αλληλεπίδρασή τους με τα αντικείμενα του περιβάλλοντος, ενώ στους δεύτερους σε περισσότερο νοητικές ικανότητες, όπως η δυνατότητα μάθησης, επικοινωνίας, εξαγωγής συμπερασμάτων. Κατά συνέπεια, οι φυσικοί πράκτορες αξιοποιούν κυρίως την έρευνα στους χώρους της ρομποτικής, συνθετικής κίνησης και μοντελοποίησης φυσικών νόμων, ενώ οι γνωσιακοί στηρίζονται σε αρχιτεκτονικές ευφυών πρακτόρων από τον χώρο της τεχνητής νοημοσύνης [23].

#### **4.3.1 Φυσικοί Πράκτορες**

Όπως είναι αναμενόμενο, το βάρος στην κατηγορία αυτή πέφτει στη φυσική συμπεριφορά.

Πρώτα απ' όλα θα πρέπει να αντιμετωπιστεί η κίνηση του σώματος και η μετακίνηση του πράκτορα στο περιβάλλον [24], ένα θέμα που επηρεάζει άμεσα τον τρόπο σχεδιασμού και αναπαράστασης του σώματος. Στο σημείο αυτό η έννοια της ενσάρκωσης είναι εξίσου σημαντική για τους εικονικούς πράκτορες, όσο και για τους πραγματικούς ανθρώπους. Εφόσον υπάρχει μια περίπλοκη φυσική αναπαράσταση, τότε υπάρχει η δυνατότητα να χρησιμοποιηθεί για πιο ρεαλιστικές αλληλεπιδράσεις με τα αντικείμενα αλλά και για ενίσχυση της επικοινωνίας με το χρήστη μέσω βλέμματος, έκφρασης προσώπου και χειρονομιών [25].

Ακόμα, καθώς οι πράκτορες έχουν τη δυνατότητα της μετακίνησης, θα πρέπει να είναι σε θέση να αποφύγουν ανεπιθύμητες συγκρούσεις με άλλα αντικείμενα στο περιβάλλον τους, είτε αυτά είναι στατικά, όπως δέντρα, κτίρια και έπιπλα, είτε κινούμενα, όπως άλλοι πράκτορες. Η ύπαρξη παραπάνω πρακτόρων στο περιβάλλον δημιουργεί ζητήματα συντονισμένης κίνησης μεγάλου αριθμού ατόμων [26][27], ενώ θα πρέπει επίσης να αντιμετωπιστούν και άλλα είδη αλληλεπίδρασης, που περιλαμβάνουν επαφή με το περιβάλλον πέρα από απλές συγκρούσεις, όπως το πιάσιμο των αντικειμένων ή οι φυσικές αλληλεπιδράσεις με άλλους πράκτορες, όπως η χειραψία.

Τέλος, με δεδομένη την ύπαρξη μιας πολύπλοκης φυσικής αναπαράστασης και ενός ρεπερτορίου φυσικών συμπεριφορών, προκύπτουν διάφορα θέματα ελέγχου. Για παράδειγμα, θα πρέπει να καθοριστεί το επίπεδο στο οποίο θα πρέπει να ασκηθεί ο έλεγχος, που περιλαμβάνει την πιθανότητα ελέγχου στο επίπεδο των ξεχωριστών μυών ή στο επίπεδο των ολοκληρωμένων ενεργειών συμπεριφοράς [15], όπως το βάδισμα και το πιάσιμο των αντικειμένων ή ακόμα και στα δύο επίπεδα. Το θέμα αυτό είναι ήδη γνωστό από το χώρο της ρομποτικής, και συγκεκριμένα από την έρευνα στον τηλεχειρισμό. Ένα δεύτερο αντίστοιχο θέμα είναι το πώς θα συνδυαστούν οι φυσικές συμπεριφορές έτσι, ώστε ο πράκτορας να είναι σε θέση να κινείται και να πιάνει αντικείμενα ταυτόχρονα.

Παρόλο που στην έρευνα των φυσικών πρακτόρων υπάρχουν πολλά κοινά ζητήματα με το χώρο της ρομποτικής, υπάρχουν και ορισμένες ενδιαφέρουσες διαφορές. Για παράδειγμα, τα προβλήματα που αφορούν στις δυνάμεις της φυσικής και στα διάφορα υλικά ή ακόμα περισσότερο στο χειρισμό μοχλών και μηχανισμών είναι πολύ πιο δύσκολα και πιο ακριβά στην επίλυσή τους σε σχέση με τους εικονικούς πράκτορες. Έτσι, οι εικονικοί πράκτορες μπορούν να κατασκευαστούν με πολύ πιο ενδιαφέρουσες δομές σε σχέση με τα περισσότερα ρομπότ. Επιπλέον οι εικονικοί αισθητήρες δε χρειάζεται να πάσχουν από τα προβλήματα που υπάρχουν

στους αληθινούς. Οι εικονικοί πράκτορες μπορούν να ξέρουν πάντα με ακρίβεια τη θέση τους στον κόσμο, αντίθετα με τα ρομπότ, για τα οποία η εύρεση της θέσης τους είναι ένα μεγάλο πρόβλημα. Από την άλλη μεριά, στον πραγματικό κόσμο η βαρύτητα, η τριβή, η αδράνεια και όλες οι άλλες φυσικές ιδιότητες είναι διαθέσιμες ελεύθερα, ενώ σε έναν εικονικό κόσμο η φυσική θα πρέπει να προστεθεί από το σχεδιαστή [27].

#### 4.3.2 Γνωσιακοί Πράκτορες

Τα θέματα που σχετίζονται με το σχεδιασμό και την ανάπτυξη των γνωσιακών μπορούν να ομαδοποιηθούν σε τρεις περιοχές κλειδιά.

Πρώτον, θα πρέπει να υπάρχει κάποιο τμήμα του πράκτορα που είναι υπεύθυνο για τις σημαντικές γνωσιακές ικανότητες της συλλογιστικής, ανάληψης αποφάσεων, σχεδιασμού ενεργειών, εκμάθησης, κλπ, ανεξάρτητα από το περιβάλλον στο οποίο βρίσκεται [28]. Αυτό είναι ακριβώς το ίδιο πρόβλημα που αντιμετωπίζεται από τους ευφυείς πράκτορες και σε άλλες περιπτώσεις, και μπορεί να θεωρηθεί ως η βάση πάνω στην οποία στηρίζονται τα υπόλοιπα τμήματα της αρχιτεκτονικής ενός γνωσιακού πράκτορα.

Το δεύτερο βασικό θέμα αφορά στο ρεαλισμό του πράκτορα στο περιβάλλον του, σε ότι έχει όμως να κάνει με τη συμπεριφορά του και όχι με την εμφάνιση και την κίνησή του. Για να είναι πρακτικά τα ευφυή εικονικά περιβάλλοντα, θα πρέπει να είναι αληθοφανή, τόσο σε ότι αφορά τις ενέργειες των πρακτόρων, όσο και στις αλληλεπιδράσεις τους με άλλους πράκτορες ή χρήστες. Αυτό πρακτικά σημαίνει ότι το γνωσιακό τμήμα δε θα πρέπει να είναι αποκομμένο από τα στοιχεία που επηρεάζουν τις αποφάσεις, δηλαδή κίνητρα, συναισθήματα, προσωπικότητα, τα οποία θα πρέπει να είναι σε θέση να εκφραστούν στο εικονικό περιβάλλον. Έχει γίνει αρκετή έρευνα στο χώρο των ευφυών πρακτόρων με συναισθήματα και σε μοντέλα συναισθημάτων και κινήτρων. Παραδείγματα είναι η εργασία των Moffat και Fridja στην ανάπτυξη ενός υπολογιστικού μοντέλου συναισθημάτων [29] και η πιο περίπλοκη «υπολογιστική θεωρία του μυαλού», που αναπτύχθηκε για πολλά χρόνια από τον Sloman [30].

Το τρίτο θέμα είναι η αναπαράσταση του πράκτορα και των ενεργειών του στον κόσμο, που ολοκληρώνει τον κύκλο, καθώς επιστρέφει στο θέμα της μοντελοποίησης γνωσιακών και συναισθηματικών μοντέλων σε φυσικά. Η οπτικοποίηση των πρακτόρων είναι σημαντική και μπορεί να είναι αποδοτική μόνο, εφόσον τα μοντέλα πρακτόρων παρέχουν αρκετά λεπτομερείς πληροφορίες. Για παράδειγμα ο [15] περιγράφει τους τρόπους με τους οποίους η προσωπικότητα, που εκδηλώνεται με διάφορους τρόπους, όπως για παράδειγμα η περιέργεια ή η κούραση, μπορεί να συσχετιστεί με διαφόρους παραμέτρους της κίνησης ενός εικονικού πράκτορα, όπως ταχύτητα και πρόβλεψη. Περιγράφουν ένα απλό μοντέλο στο οποίο η προσωπικότητα μπορεί να επηρεάσει τις ενέργειες του πράκτορα.

#### 4.3.3 Ευφυής Πράκτορας

Είναι αρκετά δύσκολο να ορίσουμε τι ακριβώς είναι αυτό που κάνει έναν πράκτορα «ευφυή». Ωστόσο, μπορούμε γενικά να πούμε ότι ένας πράκτορας είναι ευφυής όταν έχει την ικανότητα να επιτελεί τους στόχους και τα καθήκοντα που έχει επιφορτιστεί, καθώς επίσης να προσαρμόζεται και να μαθαίνει τα ενδιαφέροντα του χρήστη, ώστε να του προσφέρει μια ουσιαστική υπηρεσία. Έτσι, σε ένα ελάχιστο επίπεδο νοημοσύνης μπορεί να δίνονται στον πράκτορα εντολές με τη μορφή κανόνων και αυτός να ενεργεί με τη βοήθεια κάποιου μηχανισμού εξαγωγής συμπεράσματος. Σε ένα ανώτερο επίπεδο ο πράκτορας θα είναι ικανός να μαθαίνει και να προσαρμόζεται αυτόματα στο περιβάλλον έτσι, ώστε να πετυχαίνει τους σκοπούς του. Για αυτόν τον λόγο μπορούμε να πούμε ότι ένας πράκτορας είναι ευφυής όταν συνδυάζει τεχνικές που παρέχουν αυτές τις δυνατότητες, δηλαδή κυρίως τεχνικές από τους χώρους της Τεχνητής ή Υπολογιστικής Νοημοσύνης ( Artificial or Computational Intelligence).

Ο όρος «συμπεριφορά» στο πεδίο της τεχνητής νοημοσύνης αναφέρεται σε ένα σύστημα ελέγχου που βασίζεται στις αισθήσεις [31], στο οποίο τα εισερχόμενα ερεθίσματα αντιστοιχίζονται σε εξερχόμενες αντιδράσεις. Στο χώρο αυτό, οι αρχιτεκτονικές συμπεριφορές

στηρίζονται σε αρχιτεκτονικές ευφυών πρακτόρων [32], προσαρμοσμένες στο περιβάλλον με τους κατάλληλους αισθητήρες και επιδραστής.

Ένας ευφυής πράκτορας θα πρέπει να έχει ικανότητα αντίδρασης στα ερεθίσματα που δέχεται, επομένως πρέπει να έχει τη δυνατότητα να αντιλαμβάνεται το περιβάλλον, αλλά και να επιδρά πάνω σ' αυτό. Αυτό συμβαίνει με τη χρήση των αισθητήρων και επιδραστήων, που έχουν ήδη αναφερθεί.

Εκτός από τη δυνατότητα να αντιλαμβάνεται το περιβάλλον, στις περισσότερες περιπτώσεις ένας πράκτορας χρειάζεται να έχει μία εσωτερική κατάσταση, η οποία μπορεί να επηρεάζει τις ενέργειες και τη συμπεριφορά του. Η εσωτερική αυτή κατάσταση είναι ένα σύστημα λήψης αποφάσεων. Στην περίπτωση που ο πράκτορας δεν έχει εσωτερική κατάσταση, έχουμε μία απλουστευμένη αρχιτεκτονική, όπου οι ενέργειές του εξαρτώνται αποκλειστικά από τα ερεθίσματα που δέχεται από το περιβάλλον.

#### 4.4 Άλλες Κατηγορίες Πρακτόρων

Εκτός από τους εικονικούς πράκτορες υπάρχουν και άλλα, παραπλήσια είδη συνθετικών χαρακτήρων σε εικονικά περιβάλλοντα που συναντώνται στη διεθνή βιβλιογραφία, καθώς η έρευνα είναι σε πολλά σημεία κοινή.

Τα είδη αυτά είναι:

- Εικονικός ηθοποιός (virtual actor) [33]: Διαφέρει από έναν εικονικό πράκτορα στο ότι δεν έχει αυτονομία. Όλες οι ενέργειές του είναι προκαθορισμένες από κάποιο σενάριο ή βασίζονται σε σχετικές εντολές του χρήστη [34].
- Ενσάρκωση (avatar): Έχει οριστεί από τον ως εκδήλωση του εαυτού σε έναν εικονικό κόσμο και έχει σχεδιαστεί για να ενισχύσει την αλληλεπίδραση σε έναν εικονικό χώρο. Τα avatars επιτρέπουν στον χρήστη να λάβει μια ορατή μορφή στον εικονικό κόσμο, παρέχοντάς του την ευκαιρία να συμμετάσχει σε σουρεαλιστικές και φανταστικές εμπειρίες που υπερβαίνουν το πραγματικό κόσμο. Με λίγα λόγια, εξυπηρετεί ως εικονικός εκπρόσωπος ενός πραγματικού ανθρώπου και οι ενέργειές του ελέγχονται (σε πραγματικό χρόνο) από το άτομο αυτό. Συνεπώς, εκτός από έλλειψη αυτονομίας, χαρακτηρίζεται και από έλλειψη αισθητήρων, καθώς η αντίληψη του χώρου γίνεται από τον ίδιο το χρήστη.
- Εικονικός άνθρωπος (virtual human) [35][36]: Είναι μια ειδική περίπτωση εικονικού πράκτορα στην οποία έχει δοθεί έμφαση στη λεπτομερή μοντελοποίηση και προσομοίωση της ανθρώπινης κίνησης και αλληλεπίδρασης. Συνήθως οι χαρακτήρες αυτοί έχουν ιδιαίτερα περίπλοκους αισθητήρες και επιδραστής αλλά χαμηλό επίπεδο αυτονομίας.
- Ορθολογικός πράκτορας είναι ένας πράκτορας που ενεργεί έτσι ώστε να επιτυγχάνει το καλύτερο αποτέλεσμα ή, όταν υπάρχει αβεβαιότητα, το καλύτερο αναμενόμενο αποτέλεσμα.



## Κεφάλαιο 5 - Εικονικός Κόσμος

Ο εικονικός κόσμος προκύπτει ως αποτέλεσμα της λειτουργίας του εικονικού περιβάλλοντος ή της εικονικής πραγματικότητας και αποτελεί την αναπαράσταση ενός κόσμου με τη χρήση ηλεκτρονικών υπολογιστών. Ένας κόσμος νοείται ως η αναπαράσταση ενός συνόλου αντικειμένων, οπότε ένας εικονικός κόσμος ορίζεται ως ένα σύνολο εικονικών αντικειμένων. Τα εικονικά αντικείμενα μπορεί να αναπαριστούν οτιδήποτε από ζωντανές οντότητες μέχρι και άψυχα αντικείμενα. Επιπλέον, αντικείμενα ενός εικονικού κόσμου αποτελούν και οι εικονικοί πράκτορες που δρουν σε αυτόν ή η αναπαράσταση του ίδιου του ανθρώπου-χρήστη (avatar).

Η αναπαράσταση αποτελεί μια διαδικασία μέσω της οποίας δημιουργούμε ένα αντικείμενο ή σύνολο αντικειμένων που διεγείρει τις αισθήσεις του ανθρώπου με τέτοιο τρόπο, ώστε να παράγεται στον νου του η εικόνα αυτών των αντικειμένων. Η έννοια δεν αφορά μόνο τον φυσικό κόσμο, αλλά οποιονδήποτε κόσμο και παίρνει διάφορες μορφές. Η αναπαράσταση χρονολογείται από την προϊστορική περίοδο, με τις πρώτες σπηλαιογραφίες όπου οι άνθρωποι αναπαριστούσαν την καθημερινότητά τους και από την αρχαιότητα όπου η ανάπτυξη του θεάτρου, των τεχνών και των γραμμάτων οδήγησε στην αναπαράσταση κόσμων και αντικειμένων, με το ανέβασμα θεατρικών παραστάσεων, την καταγραφή μύθων ακόμα και την τέλεση θρησκευτικών τελετουργιών. Οι άνθρωποι προσπαθούσαν πάντοτε να έρθουν σε επαφή με έναν διαφορετικό κόσμο από αυτόν που βρίσκονταν. Αιχμή στην οπτική αναπαράσταση αποτέλεσε η ανακάλυψη της φωτογραφίας και βέβαια, η επακόλουθη δημιουργία της κινούμενης εικόνας (video). Τέλος, το πιο σύγχρονο μέσο αναπαράστασης αντικειμένων αποτελεί η χρήση τρισδιάστατων γραφικών σε υπολογιστές.

Η αναπαράσταση εικονικών κόσμων προσπαθεί να διεγείρει όλες τις αισθήσεις και δεν στοχεύει μόνο στην οπτική απεικόνιση. Ωστόσο, κάτι τέτοιο δεν είναι εφικτό με αποτέλεσμα να δίνεται έμφαση στην αίσθηση της όρασης και στην αληθοφανή οπτικοποίηση των αντικειμένων. Όπως προαναφέρθηκε, υπάρχουν διάφορα σύγχρονα μέσα και εργαλεία εικονικής πραγματικότητας που στοχεύουν στη διέγερση των αισθήσεων του χρήστη, όπως HDMS, απτικά γάντια κτλ. Επίσης, πρέπει για αυτό το λόγο τα αντικείμενα στον κόσμο να γίνονται αντιληπτά από άλλες οντότητες, είτε αυτές είναι αυτόνομοι πράκτορες που κινούνται στον εικονικό κόσμο είτε ο ίδιος ο ανθρώπινος παράγοντας μέσω του avatar του. Συνεπώς, κάποια αντικείμενα μπορεί να προσφέρουν και τη δυνατότητα αλληλεπίδρασης με έναν πράκτορα ή χρήστη.

Η διαφορά μεταξύ εικονικής πραγματικότητας/εικονικών περιβαλλόντων και εικονικών κόσμων είναι πως ο όρος «εικονικός κόσμος» έχει καθιερωθεί για τους διαρκείς διαδικτυακούς κοινωνικούς χώρους, δηλαδή για τα εικονικά περιβάλλοντα που είναι πολυπληθή, οι άνθρωποι τα αντιλαμβάνονται ως συνεχή στο χρόνο και τα βιώνουν με άλλους ανθρώπους σαν έναν χώρο κοινωνικής αλληλεπίδρασης. Έτσι λοιπόν, οι εικονικοί κόσμοι μπορούν να διακριθούν από τα διαδικτυακά παιχνίδια και τα MMORPGs (Massively Multiplayer Online Role Playing Games) στο ότι αποτελούν έναν τρίτο χώρο, έναν διαδικτυακό χώρο για κοινωνικοποίηση. Ιδωμένο από την αντίστροφη, τα διαδικτυακά παιχνίδια είναι ένα υποσύνολο εικονικών κόσμων, αυτών στους οποίους η δραστηριότητά τους περιστρέφεται γύρω από το παιχνίδι (gaming) [37].

Οι εικονικοί κόσμοι έχουν αρκετά κοινά χαρακτηριστικά και λειτουργίες που τους κάνουν ελκυστικούς στους χρήστες. Γενικά, είναι περιβάλλοντα εμπύθισης που θυμίζουν ηλεκτρονικά παιχνίδια, όπου οι συμμετέχοντες χρησιμοποιούν μια ενσώματη αναπαράσταση τους εντός του κόσμου (avatar) για να αφοσιωθούν σε μια ποικιλία δραστηριοτήτων σε ένα κοινόχρηστο χώρο. Το κοινόχρηστο περιβάλλον είναι ένα θεμελιώδες χαρακτηριστικό των εικονικών κόσμων και παρέχει τη βάση κατανόησης για ποιο λόγο δημιουργούν ένα ελκυστικό περιβάλλον στους χρήστες. «Ως οντότητες που κατοικούμε σε έναν τρισδιάστατο φυσικό κόσμο, ο εγκέφαλός μας έχει σχεδιαστεί για να αναμένει και να αποδέχεται ερεθίσματα που μοιάζουν με αυτά του πραγματικού κόσμου. Με άλλα λόγια, ανταποκρινόμαστε στη χωρική φύση των εικονικών κόσμων κατά κύριο λόγο επειδή μας θυμίζουν τον πραγματικό» [37].

Πρωτεύοντα χαρακτηριστικά, λοιπόν, ενός εικονικού κόσμου είναι η **πιστότητα** (believability), η **εμβύθιση** (immersion) και ο **ρεαλισμός** (realism). Η πιστότητα αφορά στην εφαρμογή φυσικών νόμων που ισχύουν και στον πραγματικό κόσμο και στην αληθοφανή αλληλεπίδραση μιας οντότητας – αυτόνομου ευφυούς πράκτορα ή avatar με τον κόσμο, ώστε η

συμπεριφορά εντός του κόσμου να είναι αξιόπιστη. Επομένως, όταν ένας χρήστης αλληλεπιδρά με τον εικονικό κόσμο αναμένει κάποια πράγματα να συμβούν με συγκεκριμένο τρόπο, για παράδειγμα η βαρύτητα πρέπει να προκαλεί τα αντικείμενα να πέφτουν και ο ήχος να εξασθενίζει όσο απομακρύνεται κανείς από την πηγή ή να δυναμώνει στην αντίθετη περίπτωση, προσομοιώνοντας τους φυσικούς νόμους του πραγματικού κόσμου στον εικονικό.

Όσο πιο πιστευτός είναι ένας εικονικός κόσμος, δηλαδή όσο πιο πιστός στους φυσικούς κανόνες του πραγματικού κόσμου, τόσο αυξάνεται και η αίσθηση παρουσίας του χρήστη σε αυτόν, η εμπύθιση, δηλαδή η αποκοπή των αισθήσεων του χρήστη από τον πραγματικό κόσμο και ο προσανατολισμός των αισθήσεων και της προσοχής του στον εικονικό κόσμο.

Τέλος, ο ρεαλισμός σχετίζεται με την οπτική αναπαράσταση του κόσμου, δηλαδή την απεικόνιση, και αφορά το βαθμό ομοιότητας του εικονικού κόσμου με τον πραγματικό. Έρευνες έχουν δείξει πως η δυνατότητα να αγγίζεις εικονικά αντικείμενα κάνει τα αντικείμενα αυτά και τον εικονικό κόσμο πολύ πιο αληθοφανή. Όταν αναμνήσεις ανάμεικτης πραγματικότητας προσεγγίζουν πραγματικές αναμνήσεις, οι άνθρωποι είναι πιο πιθανό να μπερδεύουν πραγματικά και εικονικά αντικείμενα, ανάλογα με τον βαθμό ρεαλισμού [38].

Υπάρχουν διάφορα παραδείγματα εικονικών κόσμων τα οποία μπορούμε να αναφέρουμε, όπως το World of Warcraft, Second Life, There, Active Worlds, Kaneva, EverQuest, Final Fantasy, Star Wars Universe, γνωστά και ως πολυχρηστικά εικονικά περιβάλλοντα (MUVES – Multi User Virtual Environments), καθώς επιτρέπουν την ταυτόχρονη ύπαρξη και αλληλεπίδραση πολλών χρηστών σε αυτούς. Όλοι αυτοί οι αναφερόμενοι κόσμοι έχουν τη δυναμική να αλλάζουν τον τρόπο με τον οποίο οι άνθρωποι αλληλεπιδρούν, σερφάρουν στο διαδίκτυο και διευθύνουν τις επιχειρήσεις τους. Η ραγδαία ανάπτυξη τους οφείλεται, από τη μια, στα εξελιγμένα τρισδιάστατα περιβάλλοντα που παρέχουν στους χρήστες εκπληκτικά γραφικά και, από την άλλη, στη δυνατότητα που παρέχουν στους χρήστες να υποκρίνονται φανταστικούς ρόλους και να δημιουργούν κοινότητες. Επίσης, η αλληλεπίδραση που προσφέρουν με άλλους χρήστες – φίλους ή γνωστούς, ακόμα και αγνώστους - έχει αυξήσει την δημοτικότητα τους. Όσο αυτά τα περιβάλλοντα γίνονται πιο διεισδυτικά, τόσο περισσότερη έρευνα απαιτείται για την καλύτερη κατανόηση αυτών των τρισδιάστατων χώρων. Ήδη πληροφοριακοί επιστήμονες εξερευνούν αυτό το πεδίο και μπορούν να μας διαφωτίσουν πάνω στο σχεδιασμό, την ανάπτυξη, τη διαχείριση και τη χρήση αυτών των εικονικών κόσμων [38].

## 5.1 Ο Εικονικός Κόσμος Second Life

Αξίζει να γίνει μία μικρή αναφορά στο *Second Life*, ένα ιδιαίτερο πολυχρηστικό περιβάλλον, που αποτελεί αιχμή στην εξέλιξη των εικονικών κόσμων και της εικονικής πραγματικότητας.

Το Second Life (SL) είναι ένα διαδικτυακό παιχνίδι υπολογιστή που αναπαριστά ένα εικονικό, τρισδιάστατο περιβάλλον, του οποίου η λειτουργία διαμορφώνεται εξ' ολοκλήρου από τους «κατοίκους» του. Το SL δόθηκε στο κοινό το 2003 από την εταιρεία Linden Lab. Το SL αποτελείται από έναν εικονικό κόσμο όπου οι χρήστες μπορούν να ψηφιοποιήσουν τα πάντα και να συνεργαστούν σε ένα τρισδιάστατο περιβάλλον που θα φτιαχνόταν από τους ίδιους [39].

Στο SL, που απέχει πολύ από τα συνηθισμένα πολυσυμμετοχικά διαδικτυακά παιχνίδια, δεν υπάρχουν προκαθορισμένοι κανόνες, οδηγίες και στόχοι που πρέπει να εκπληρωθούν. Το παιχνίδι δίνει τη δυνατότητα στους παίκτες του να δημιουργήσουν τον ψηφιακό τους εαυτό - avatar - και να ζήσουν μια παράλληλη πραγματικότητα μέσα από την οθόνη του υπολογιστή τους, χωρίς να υπάρχουν νικητές ή χαμένοι και πόντοι που υπάρχουν συνήθως σε άλλα ηλεκτρονικά παιχνίδια. Αναπαριστά -στο μέτρο του δυνατού- μία νέα, εναλλακτική πραγματικότητα, όπου ο καθένας μπορεί να παρουσιάσει τον εαυτό του όπως το επιθυμεί, απαλλαγμένος από τις συμβάσεις και τους περιορισμούς της πραγματικής του ζωής, ενώ έχουν δημιουργηθεί όλα αυτά που συνιστούν και τον πραγματικό κόσμο: σπίτια, νοσοκομεία, δρόμοι, πόλεις, μεταφορικά μέσα, επιχειρήσεις, θέατρα, κινηματογράφοι, οτιδήποτε μπορεί να συναντήσει κανείς και στην κανονική ζωή. Στον κόσμο του SL μπορείς να δεις καθηγητές Πανεπιστημίου που διδάσκουν τους σπουδαστές τους, ωθώντας τους να ανοίξουν εικονικά καταστήματα, μεσίτες ακινήτων που συμπληρώνουν το εισόδημα τους πουλώντας εικονικά

οικόπεδα, ακόμα και μουσικούς ή άλλους καλλιτέχνες οι οποίοι παρουσιάζουν -σχεδόν ανέξοδα- τα έργα τους στα εκατομμύρια των κατοίκων του SL [39].



**Εικόνα 25: SecondLife "όπως στην κανονική ζωή"**

Ωστόσο, η απουσία δηλωμένων στόχων είναι η ιδιαιτερότητα και το προτέρημα του SL, καθώς οι ψηφιακές απεικονίσεις των χρηστών (avatars) είναι σε θέση να κάνουν νέες κοινωνικές γνωριμίες και να εργάζονται κανονικά, είτε ως υπάλληλοι σε εικονικές εταιρίες, είτε προωθώντας τις δικές τους επιχειρηματικές ιδέες και τελικά να αποκτούν χρήματα (στην αρχή εικονικά και μετέπειτα πραγματικά), μέσω μιας πλήρως αυτόνομης οικονομίας που ανταμείβει την καινοτομία, τη δεξιοτεχνία και το ρίσκο. Επομένως, το παιχνίδι δίνει τη δυνατότητα στους χρήστες να εργαστούν και να επενδύσουν σε αυτό, αφού λειτουργεί με τα επιχειρηματικά κριτήρια της αγοράς που υπάρχουν και στην πραγματικότητα [39].



**Εικόνα 26: Second Life στο Βερολίνο**

Το παιχνίδι έχει ακόμα και το δικό του νόμισμα, το δολάριο Linden, που μπορεί ο οποιοσδήποτε παίχτης να το εξαργυρώσει με κανονικά χρήματα. Οι κάτοικοι μπορούν να αγοράσουν απευθείας Linden ή να κάνουν ανταλλαγές συναλλάγματος μεταξύ Linden και αμερικανικού δολαρίου, το οποίο και αποταμιεύεται στον πραγματικό τραπεζικό λογαριασμό τους, μέσω του συστήματος LindeX. Η ισοτιμία του δολαρίου Η.Π.Α. και Linden μεταβάλλεται συνεχώς, καθώς οι κάτοικοι του SL, μέσω των επιχειρηματικών και οικονομικών συναλλαγών τους, καθορίζουν την τιμή αγοράς και πώλησης των δολαρίων Linden. Αξίζει να σημειωθεί ότι τον Ιανουάριο του 2008 η αξία ενός αμερικανικού δολαρίου ήταν 267 L\$, ενώ το συνολικό ΑΕΠ του SL άγγιξε τα 65 εκατομμύρια αμερικανικά δολάρια.

Βέβαια, καθώς πίσω από το Second Life κρύβονται πραγματικοί άνθρωποι, στον εικονικό κόσμο του παιχνιδιού έχουν παρατηρηθεί ορισμένα από τα φαινόμενα που υπάρχουν και στον πραγματικό κόσμο, όπως η ξеноφοβία, οι λεκτικές και σωματικές επιθέσεις, η βία, οι κοινωνικές αντιδράσεις και η κοινωνική απομόνωση.

Το βέβαιο είναι πως εικονικοί κόσμοι όπως το SL, αποτελούν μια νέα δύναμη στο χώρο της ηλεκτρονικής εκπαίδευσης. Σε σύγκριση με το game-based learning που αναφέρεται σε μικρή ομάδα ατόμων που εργάζονται απομονωμένα με περιορισμένες πηγές, η εταιρία Linden Lab εκμεταλλεύτηκε βασικές έννοιες του Web 2.0, όπως την συνεργατική δημιουργία (creative commons) και το λογισμικό ανοικτού κώδικα (open source), και δημιούργησε ένα εικονικό κόσμο στον οποίο εκπαιδευτικές προσομοιώσεις και προγράμματα μάθησης μπορούν σχετικά εύκολα και οικονομικά να εφαρμοστούν [39].

## **5.2 Εφαρμογές Των Εικονικών Κόσμων**

Φαίνεται, λοιπόν, πως οι εικονικοί κόσμοι είναι ιδιαίτερα χρήσιμοι και αποτελούν ισχυρά εκπαιδευτικά εργαλεία καθώς: πρώτον, δίνουν τη δυνατότητα στους χρήστες τους να εκτελούν καθήκοντα που μπορεί να είναι δύσκολα για αυτούς στον πραγματικό κόσμο εξαιτίας διάφορων περιορισμών (πχ κόστος, χρονοπρογραμματισμός, τοποθεσία), δεύτερον, η διάρκεια (στο χρόνο) των εικονικών κόσμων επιτρέπει τη διατήρηση και την καλλιέργεια κοινωνικών σχέσεων που μπορούν να λειτουργήσουν ως βάση για τη συμμετοχική εκπαίδευση και τρίτον, οι εικονικοί κόσμοι μπορούν να επεκταθούν και να προσαρμοστούν στις ανάγκες του χρήστη.

Η συσχέτιση με εικονικούς κόσμους επιτρέπει στους μαθητές να βιώσουν μαθησιακές ευκαιρίες που δε θα ήταν εύκολα προσιτές, όπως το παιχνίδι ρόλων, ο χειρισμός προσομοιωμένου εξοπλισμού, η σχεδίαση και η δημιουργία πραγμάτων και η δημιουργία προσομοιώσεων για φυσικές διαδικασίες. Μέσα από αυτές τις δραστηριότητες οι μαθητές καλλιεργούν τη γνωστική τους λειτουργικότητα, όπως είναι η ερμηνεία, η κατανόηση, η ανάλυση, η ανακάλυψη, η σύγκριση, ο υπολογισμός και η επίλυση προβλημάτων [40].





**Εικόνα 27: Avatars στο Lineage**

Οι εικονικοί κόσμοι, ωστόσο, διαφέρουν μεταξύ τους. Υπάρχουν κατηγορίες όπως το SecondLife που εντρυφούν περισσότερο στην κοινωνικοποίηση και «τη δυνατότητα να ζήσεις μια ζωή όπως πραγματικά την ήθελες» [40]. Από την άλλη, κόσμοι όπως το Lineage έχουν περισσότερο χαρακτήρα ηλεκτρονικού παιχνιδιού με το χρήστη να υποδύεται ρόλους. Άλλοι κόσμοι, όπως το Virtual Battle field System One, έχουν σχεδιαστεί για τον στρατό και την προσομοίωση ασκήσεων μάχης και διάσωσης πολιτών.



**Εικόνα 28: Virtual Battle field System One**

Άλλοι απευθύνονται σε ανθρώπους μεγάλης ηλικίας ή βετεράνους πολέμου, για να τους βοηθήσουν να ξεπεράσουν τυχόν ψυχικά τραύματα. Τέλος, υπάρχουν και κόσμοι που απευθύνονται σε νεαρές ηλικίες, μόλις 8-13 ετών και εστιάζουν στη διαδικασία της μάθησης [40].

Στα πλαίσια αυτής της μεταπτυχιακής διατριβής, ο εικονικός κόσμος απεικονίζεται σε ηλεκτρονική μορφή στη δισδιάστατη οθόνη ενός υπολογιστή, σχεδιάζεται στην xml-based

γλώσσα `verl` και αναπαριστά τη σύνθεση ενός συνόλου τρισδιάστατων αντικειμένων, σχεδιασμένων σε γλώσσα VRML/X3D [40].

## Κεφάλαιο 6 - Γλώσσες Αναπαράστασης Εικονικών Κόσμων

Υπάρχουν διάφορες γλώσσες που χρησιμοποιούνται για την αναπαράσταση εικονικών κόσμων, η καθεμία με τα δικά της ιδιαίτερα χαρακτηριστικά. Σε αυτό το κεφάλαιο θα δούμε μια συνοπτική περιγραφή για τη γλώσσα χαμηλού επιπέδου OpenGL, καθώς και τι είναι ένα γράφημα σκηνής και ποιες οι γλώσσες υψηλού επιπέδου που χρησιμοποιούν γραφήματα σκηνής: VRML, X3D και Java 3D.

### 6.1 Η Γλώσσα Τρισδιάστατων Γραφικών OpenGL

Η γλώσσα OpenGL (Open Graphics Library) χρησιμοποιείται για σχεδίαση 2D και 3D γραφικών χαμηλού επιπέδου, με την έννοια ότι μιλάει απευθείας στην κάρτα γραφικών. Με τον όρο OpenGL δεν αναφερόμαστε σε μια συγκεκριμένη βιβλιοθήκη, αλλά σε ένα *πρότυπο υλοποίησης βιβλιοθηκών σχεδίασης γραφικών*. Το πρότυπο καθορίζει μια *προγραμματιστική διεπιφάνεια* (Application Programming Interface). Εμπεριέχει, δηλαδή, το σύνολο των συναρτήσεων που πρέπει να υλοποιεί μία βιβλιοθήκη γραφικών προκειμένου να είναι συμβατή με αυτό. Το API χρησιμοποιείται για να αλληλεπιδράσει με την GPU (Graphics Processing Unit) και να εκτελέσει το rendering με τη βοήθεια των επιταχυντών, όπως είδαμε παραπάνω.

Η OpenGL αναπτύχθηκε από την εταιρία Silicon GraphicsInc. στις αρχές του 1990 και έκτοτε χρησιμοποιείται ευρέως στις εφαρμογές CAD, στην εικονική πραγματικότητα, σε προσομοιώσεις πτήσεων, σε επιστημονικές εφαρμογές για 3D οπτικοποίηση δεδομένων και σε βιντεοπαιχνίδια. Την OpenGL σήμερα τη διαχειρίζεται ένας μη κερδοσκοπικός συνεταιρισμός ονόματι KhronosGroup [41].

Εφόσον με τον όρο OpenGL δεν αναφερόμαστε σε μια συγκεκριμένη βιβλιοθήκη αλλά σε ένα πρότυπο που ορίζει τη λειτουργικότητα μιας βιβλιοθήκης σχεδίασης, μπορούμε να ακολουθήσουμε τις ίδιες συμβάσεις σε όλες τις υλοποιήσεις του προτύπου (τις ίδιες εντολές). Αυτό σημαίνει δυο πράγματα: α) δεν υπάρχει περιορισμός ως προς τη γλώσσα προγραμματισμού στην οποία θα υλοποιηθεί το πρότυπο της OpenGL. Ενδεικτικά, μπορεί να χρησιμοποιηθούν βιβλιοθήκες της σε γλώσσες προγραμματισμού Fortran, C, C++, Java, Python κ.ά. και β) ο κώδικας που συντάσσουμε είναι ανεξάρτητος πλατφόρμας (platform independent) και μπορεί να εκτελεστεί σε ευρεία γκάμα λειτουργικών συστημάτων (Windows, Linux, MacOS) χωρίς ριζική τροποποίηση της δομής του. Οι βιβλιοθήκες των περισσότερων νέων μεταγλωπτιστών εμπεριέχουν (ή υπάρχει η δυνατότητα να ενσωματωθεί σε αυτούς) μια υλοποίηση της OpenGL. Αυτά είναι και τα δυο μεγαλύτερα πλεονεκτήματα που προσφέρει η OpenGL σε σύγκριση με άλλα API [41].

Η αρχιτεκτονική της OpenGL δουλεύει κυρίως με καταστάσεις (states), δηλαδή ο προγραμματιστής ορίζει μια κατάσταση και η OpenGL κάνει render με βάση αυτή την κατάσταση μέχρι να γίνει αλλαγή της τρέχουσας κατάστασης. Για παράδειγμα, εάν θέλω να κάνω render με κόκκινο χρώμα, θέτω το χρώμα ως κόκκινο και κάνω render τα αντικείμενα που θέλω με αυτό το χρώμα.

Οι κατηγορίες βιβλιοθηκών που συναντά κανείς σε υλοποιήσεις της OpenGL είναι οι:

- OpenGL Core Library
- OpenGL Utility Library (GLU)
- OpenGL Utility Toolkit (GLUT)

Η βασική (core) βιβλιοθήκη έχει σχεδιαστεί ως μια βελτιωμένη διεπαφή ανεξάρτητη από το hardware και είναι η:

- OpenGL32 για Windows
- GL για Unix / Linux συστήματα

Η GLU παρέχει πολλά από τα χαρακτηριστικά μοντέλων, όπως quadric επιφάνειες και NURBS καμπύλες και επιφάνειες. Παρέχει λειτουργικότητα στην βασική OpenGL αλλά δεν χρειάζεται να ξαναγράψουμε κώδικα.

Η GLUT συνδέει την OpenGL με το παραθυρικό σύστημα:

- GLX για X συστήματα
- WGL για Windows
- AGL για Macintosh

## 6.2 Γλώσσα Προγραμματισμού VRML

Η VRML (Virtual Reality Modeling Language) είναι μια γλώσσα προγραμματισμού για τη δημιουργία τρισδιάστατων (3D) γραφικών αλληλεπίδρασης. Η VRML οφείλει την προέλευσή της στο πρώτο παγκόσμιο συνέδριο δικτύου το Μάρτιο του 1994, αναπτύχθηκε από το Web3D Consortium και προτυποποιήθηκε από την ISO/IEC. Η VRML χρησιμοποιείται κατά κόρον για τρισδιάστατα γραφικά στο διαδίκτυο, οπότε οι προγραμματιστές μπορούν να δημιουργήσουν ιστοσελίδες με ακολουθίες εικόνων, έτσι ώστε οι επισκέπτες να μπορούν να αλληλεπιδράσουν με αυτές, να τις περιστρέφουν ή να περιηγούνται σε αυτές [42][43].

Ουσιαστικά, η γλώσσα VRML είναι το πρότυπο 3D γραφικών για τον παγκόσμιο ιστό (www) και δημιουργεί ένα συγκεκριμένο τύπο αρχείου, το οποίο περιέχει μια ASCII περιγραφή τρισδιάστατων σκηνών (γράφημα σκηνής). Η συγγραφή προγραμμάτων σε VRML απαιτεί την ύπαρξη ενός οποιουδήποτε tex editor και την αποθήκευση των αρχείων με την κατάληξη .wrl (από το world=κόσμος). Οι χαρακτήρες που επιτρέπονται καθορίζονται από το UTF-8 format, του οποίου υποσύνολο είναι οι ASCII χαρακτήρες. Τα προγράμματα παρουσιάζονται σε οποιονδήποτε browser, με την προϋπόθεση ότι έχει εγκατασταθεί το απαιτούμενο plugin για VRML 2 στον υπολογιστή. Με την VRML μπορούμε να σχεδιάσουμε τρισδιάστατα γεωμετρικά αντικείμενα από τα οποία αποτελείται ένας εικονικός κόσμος και να συνθέσουμε εξολοκλήρου έναν εικονικό κόσμο. Επιπλέον, επιτρέπει τον ορισμό διαφόρων σημείων-συνδέσμων, τα οποία οδηγούν σε κάποιο συγκεκριμένο URL [42][43].

Η γενική δομή της γλώσσας έχει ως εξής: Τα VRML αρχεία αποτελούνται από την επικεφαλίδα (header), τα σχόλια (comments) του προγραμματιστή, τους ορισμούς κόμβων (nodes), που μάλιστα είναι το θεμελιώδες δομικό στοιχείο της γλώσσας και φυσικά, από τα ονόματα των κόμβων, τα πεδία (fields) τους και τις αντίστοιχες τιμές τους. Με μια πιο αφαιρετική ματιά, τα VRML αρχεία αποτελούνται από σύνολα κόμβων που είναι ενταγμένα σε ιεραρχίες υποσυνόλων και αλληλεπιδρούν μεταξύ τους, δημιουργώντας ένα γράφημα σκηνής. Τα σύνολα αυτά αποτελούν τον κόσμο (world). Οι κόμβοι χρησιμοποιούνται για να εκφράσουν τα σχήματα και τις αντίστοιχες ιδιότητες τους, ενώ κάποια γεωμετρικά σχήματα προσφέρονται ως έτοιμα αντικείμενα (πχ Box, Sphere, Cylinder). Κατά τον ορισμό των κόμβων απαιτείται να καθορίζεται το όνομα του κόμβου, το είδος του, καθώς και ένας αριθμός προαιρετικών πεδίων με τις αντίστοιχες τιμές τους. Ως τιμές των πεδίων μπορεί να είναι μεταξύ άλλων και κάποιος άλλος κόμβος. Επίσης, τα πεδία μπορεί να ορίζουν πώς κόμβοι του ίδιου τύπου διαφέρουν μεταξύ τους, καθώς και να ορίζουν μια σειρά γεγονότων (events) που κάποιος κόμβος μπορεί να αποστείλει ή να εκλάβει. Όταν ένας κόμβος λαμβάνει ένα γεγονός, αντιδρά μεταβάλλοντας την κατάσταση του, η οποία μπορεί να πυροδοτήσει επιπρόσθετα γεγονότα. Οι κόμβοι μπορούν να αλλάξουν την κατάσταση των αντικειμένων στη σκηνή με τη χρήση γεγονότων. Επίσης, η υλοποίηση ενός κόμβου ορίζει πώς αντιδρά σε γεγονότα, τότε μπορεί να παράγει και να αποστείλει γεγονότα, καθώς και οποιαδήποτε οπτική ή ακουστική αναπαράσταση μπορεί να έχει στη σκηνή [42][43].

Ο αρχικός κόμβος ονομάζεται γονέας (parent), ενώ όλοι οι άλλοι κόμβοι που περιέχονται σε αυτόν αποκαλούνται παιδιά του (children). Προφανώς ο κόμβος-γονέας ή αλλιώς πρόγονος των υπολοίπων μπορεί να είναι παιδί σε κάποιον άλλο γονέα. Κατ' αυτό τον τρόπο φτάνουμε σε έναν αρχικό κόμβο από τον οποίο, τελικά, προκύπτουν όλοι οι υπόλοιποι, τον κόμβο ρίζα (root node). Ένας κόμβος μπορεί να οριστεί μια φορά και να χρησιμοποιηθεί περισσότερες φορές χωρίς να οριστεί, με τη χρήση των λέξεων DEF και USE, όπου η πρώτη καθορίζει

το όνομα του κόμβου που θα χρησιμοποιηθεί ξανά και η δεύτερη τον αναφέρει όταν χρησιμοποιείται. Επίσης, υπάρχουν κόμβοι που χρησιμοποιούνται για να ομαδοποιούν αντικείμενα, όπως ο κόμβος Group [42][43].

Μπορούμε να κατηγοριοποιήσουμε γενικά τους κόμβους που ορίζει το πρότυπο VRML ως εξής:

- Γεωμετρικοί κόμβοι που ορίζουν το σχήμα ή τη μορφή ενός αντικειμένου (πχ Shape)
- Κόμβοι γεωμετρικών ιδιοτήτων που χρησιμοποιούνται για να ορίσουν συγκεκριμένες όψεις των γεωμετρικών κόμβων
- Κόμβοι εμφάνισης (appearance) που ορίζουν το υλικό της γεωμετρίας και ιδιότητες υψής για το αντικείμενο
- Κόμβοι ομαδοποίησης που ορίζουν ένα χώρο συντεταγμένων για κόμβους-παιδιά που μπορεί να εμπεριέχουν. Για παράδειγμα, ένας κόμβος ομαδοποίησης που περιέχει έναν ή περισσότερους κόμβους-παιδιά είναι ο Transform. Κάθε Transform κόμβος έχει το δικό του τοπικό σύστημα συντεταγμένων πάνω στο οποίο τοποθετούνται οι κόμβοι-παιδιά, που μπορεί να είναι άλλοι Transform κόμβοι, Group κόμβοι ή Shape κόμβοι. Επίσης, ο κόμβος Transform υποστηρίζει πράξεις μετασχηματισμού των αντικειμένων που αφορούν τη θέση στο χώρο, την κλίμακα και το μέγεθος και εφαρμόζονται σε όλα τα παιδιά του κόμβου.
- Κόμβοι για πηγές φωτός(light-source) που φωτίζουν αντικείμενα στη σκηνή
- Κόμβοι αισθητήρων (sensornodes) που αντιδρούν σε αλλαγές του περιβάλλοντος και ενέργειες του χρήστη
- Interpolator κόμβοι που ορίζουν λειτουργίες για animations
- Κόμβοι χρονο-εξάρτησης που ενεργοποιούνται και απενεργοποιούνται σε συγκεκριμένα διαστήματα του χρόνου
- Δεσμευόμενοι (bindable) κόμβοι-παιδιά που είναι μοναδικοί, γιατί μόνο ένας κόμβος κάθε είδους μπορεί να δεσμευτεί ή να επηρεάσει την εμπειρία του χρήστη μια συγκεκριμένη χρονική στιγμή
- Script κόμβοι που διευκολύνουν τη δυναμική συμπεριφορά ενός εικονικού περιβάλλοντος, επιτρέποντας τη χρήση προγραμματιστικών γλωσσών, όπως η ECMAScript, η JavaScript, και η Java. Οι Script κόμβοι χρησιμοποιούνται κυρίως για να υποδηλώσουν μια αλλαγή στη σκηνή ή κάποιου είδους ενέργεια του χρήστη, για να εκλάβουν γεγονότα (events) από άλλους κόμβους, για να ενθυλακώσουν modules του προγράμματος που παράγουν υπολογισμούς ή για να πυροδοτήσουν αλλού στη σκηνή αλλαγές με αποστολή γεγονότων. Ο εξωτερικός προγραμματιστικός έλεγχος πάνω στο γράφημα σκηνής της VRML γίνεται μέσω του External Authoring Interface (EAI), που είναι ένα μοντέλο διασύνδεσης της διεπαφής μεταξύ των κόσμων της VRML και εξωτερικών περιβαλλόντων.

Η γλώσσα VRML σχεδιάστηκε για ικανοποιεί τις παρακάτω ανάγκες και απαιτήσεις: [44]

- **Authorability** (συγγραφικότητα): Η VRML πρέπει να καθιστά δυνατή την ανάπτυξη συστημάτων εφαρμογών και επεξεργαστών κειμένου (editor), καθώς και την εισαγωγή δεδομένων από άλλους τύπους αρχείων (format).
- **Completeness** (ολοκλήρωση): Να υπάρχουν όλες οι πληροφορίες που χρειάζονται για τον χειρισμό και να υπάρχει ένα ολοκληρωμένο σύνολο χαρακτηριστικών.
- **Composability** (συνθεσιμότητα): Η δυνατότητα να συνδυάζονται τα στοιχεία της VRML και να καθίστανται επαναχρησιμοποιήσιμα.
- **Extensibility** (επεκτασιμότητα): Η δυνατότητα να προστίθενται νέα στοιχεία.
- **Implementability** (υλοποιησιμότητα): Οι προδιαγραφές της VRML πρέπει να παρέχουν επαρκείς πληροφορίες για την υλοποίηση χωρίς να υπάρχουν αμφισημίες.



- **Κανότητα Multi –User:** Να μην αποκλείει την υλοποίηση σε πολυχρηστικά περιβάλλοντα.
- **Orthogonality** (ορθογωνικότητα): Τα στοιχεία VRML πρέπει να είναι ανεξάρτητα μεταξύ τους, αλλά στην περίπτωση που υπάρχουν εξαρτήσεις, αυτές πρέπει να είναι καλά ορισμένες και δομημένες.
- **Performance** (απόδοση): Τα στοιχεία πρέπει να σχεδιάζονται με έμφαση στη διαδραστική απόδοση πάνω σε διαφορετικές πλατφόρμες υπολογιστών.
- **Scalability:** Τα στοιχεία της VRML είναι σχεδιασμένα για μεγάλες συνθέσεις, αλλά πρέπει να έχουν περιορισμένο πεδίο εφαρμογής.
- **Well - structured** (καλά δομημένη): Τα στοιχεία πρέπει να έχουν μια καλά ορισμένη διεπαφή και έναν απλά ορισμένο σκοπό. Στοιχεία πολλαπλών σκοπών και παρενέργειες πρέπει να αποφεύγονται.



**Εικόνα 29: Το τσαγιερό της Utah. Πρωτοκατασκευάστηκε στο πανεπιστήμιο της Γιούτα το 1975 από τον M.Newell. Το πολυγωνικό πλέγμα τσαγιερού έχει γίνει ένα σημαντικό στοιχείο στα κομπιούτερ γραφικών. Αυτή η εκδοχή του VRML περιέχει 1976 κορυφές και 3751 πολύγωνα.**

Το εύρος και το πεδίο εφαρμογής της VRML περιλαμβάνουν:

- Έναν μηχανισμό για αποθήκευση και μεταφορά δισδιάστατων και τρισδιάστατων δεδομένων
- Στοιχεία αναπαράστασης δισδιάστατης και τρισδιάστατης ακατέργαστης πληροφορίας
- Στοιχεία για τον ορισμό των ιδιοτήτων αυτής της πληροφορίας και στοιχεία για την όψη και την μοντελοποίηση δισδιάστατης και τρισδιάστατης πληροφορίας
- Έναν μηχανισμό ενσωμάτωσης δεδομένων από άλλους τύπους μετά-αρχείου
- Έναν μηχανισμό ορισμού νέων στοιχείων που επεκτείνουν τις δυνατότητες του μετά-αρχείου ώστε να υποστηρίξει επιπρόσθετους τύπους και είδη πληροφορίας.

Το μεγάλο πλεονέκτημα της VRML έγκειται στο γεγονός ότι μια περιγραφή φτάνει (μέσω του Internet) στον browser του χρήστη ως ένα απλό ASCII αρχείο και στη συνέχεια ο browser αναλαμβάνει να κάνει το λεγόμενο rendering, δημιουργεί δηλαδή μια όψη (view) της σκηνής με βάση την ASCII περιγραφή. Αυτό σημαίνει ότι συνήθως χρειάζονται λιγότερα bytes για να περιγραφεί μια σκηνή. Το σημαντικότερο, όμως, είναι ότι ο browser μπορεί να αναλάβει να δείξει την ίδια σκηνή και από μια διαφορετική οπτική γωνία. Ο χρήστης, κατά αυτόν τον τρόπο, έχει τη δυνατότητα ουσιαστικά να περιηγηθεί μέσα στη σκηνή. Τυπικά, ο χρήστης έχει την επιλογή να περπατήσει ή να πετάξει μέσα στον χώρο οποιαδήποτε στιγμή, χρησιμοποιώντας το ποντίκι ή τους πίνακες ελέγχου πάνω στον browser. Ένας χρήστης μπορεί να κοιτάξει τριγύρω ή πάνω από αντικείμενα, να περπατήσει ή να πετάξει μέσα στο περιβάλλον, να εξετάσει αντικείμενα, ακόμη και να αλληλεπιδράσει με αυτά [44].

Η τρέχουσα και πλήρης λειτουργική έκδοση της VRML αντιστοιχεί στο πρότυπο ISO/IEC 14772-1:1997 και είναι γνωστή ως VRML97. Μια σύγχρονη γλώσσα που διαδέχτηκε την VRML είναι η γλώσσα X3D (ISO/IEC 19775-1), η οποία προσφέρει σημαντικές μειώσεις στις απαιτήσεις ανάπτυξης ενώ εξελίσσει το χώρο των γραφικών 3D τόσο στο διαδίκτυο όσο και πέρα από αυτό [42][43].

### 6.3 Αρχιτεκτονική Αναπαράστασης Τρισδιάστατων Γραφικών X3D

Το X3D (eXtensible 3D) είναι μια αρχιτεκτονική που αναπαριστά τρισδιάστατα γραφικά χρησιμοποιώντας την εννοιολογική μοντελοποίηση μέσω ενός γραφήματος σκηνής. Αποτελεί την διαδοχή της VRML και μας δίνει την δυνατότητα να περιγράψουμε μία τρισδιάστατη σκηνή (γράφημα σκηνής) με ορισμούς αντικειμένων είτε σε VRML97 είτε σε XML [45].

Το X3D σχεδιάστηκε για μια πληθώρα συστημάτων – για επιστημονικά workstations, επιτραπέζιους και φορητούς υπολογιστές, PDAs, tablets, κινητά τηλέφωνα και συσκευές που δεν έχουν την απαραίτητη υπολογιστική ισχύ που απαιτείται για την VRML. Το X3D δίνει επίσης τη δυνατότητα της ενσωμάτωσης υψηλής απόδοσης 3D σε μετάδοση (broadcast) και ενσωματωμένες συσκευές, ενώ αποτελεί και τον ακρογωνιαίο λίθο των αρχικών δυνατοτήτων MPEG-4 για 3D. Η χρήση του έχει ευρύ πεδίο εφαρμογών – από τη μηχανική και την απεικόνιση επιστημονικών δεδομένων έως τα πολυμέσα, τη ψυχαγωγία, την εκπαίδευση, ακόμα και σε πολυχρηστικά περιβάλλοντα. Όπως και η VRML, το X3D έχει σχεδιαστεί ως ένα διεθνές πρότυπο για ενσωματωμένα 3D γραφικά και πολυμέσα. Ωστόσο, υποστηρίζει πολλαπλές κωδικοποιήσεις (encodings), όπως η κωδικοποίηση XML, καθιστώντας το πιο σύγχρονο και δημοφιλές πρότυπο από την VRML [45].

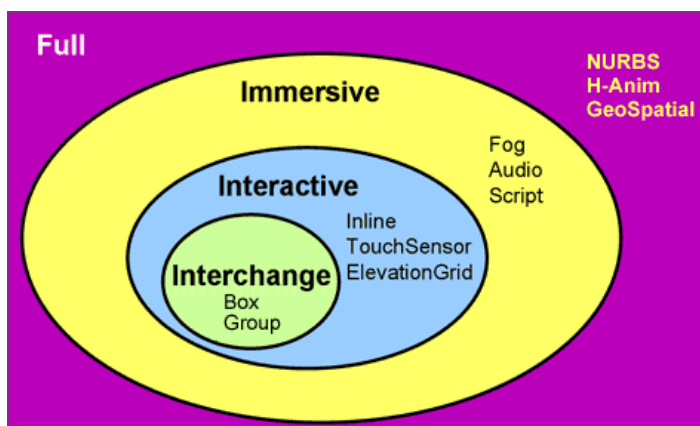
Η αρχιτεκτονική του X3D αποτελείται από αρθρωτά συστατικά στοιχεία λειτουργικότητας, γνωστά ως components, που είναι εύκολα κατανοητά και υλοποιήσιμα από τους προγραμματιστές των εφαρμογών. Γενικά, θα περιγράψαμε ένα X3D component ως ένα σύνολο από συσχετισμένες λειτουργίες, αποτελούμενες από διάφορα αντικείμενα και υπηρεσίες (services). Τυπικά πρόκειται για μια συλλογή κόμβων. Βέβαια, ένα component μπορεί να ενέχει και κωδικοποιήσεις, υπηρεσίες API, ή άλλες ιδιότητες του X3D. Η πιο σημαντική διαφορά σε σχέση με την VRML είναι πως, σπάζοντας την λειτουργικότητα σε διακριτά συστατικά στοιχεία που φορτώνονται κατά το χρόνο εκτέλεσης, μειώνεται σημαντικά η πολυπλοκότητα της εφαρμογής.

Το πρότυπο X3D προδιαγράφει ένα σύνολο από components που περιλαμβάνουν ένα Core component, το οποίο ορίζει την κύρια λειτουργικότητα κατά το χρόνο εκτέλεσης για το σύστημα X3D, αφηρημένους (abstract) τύπους κόμβων, πεδία (fields), δρομολόγηση (routing) και μοντέλα γεγονότων (event model). Κάποια components που υποστηρίζει από μόνο του το πρότυπο είναι: Time (κόμβοι που παρέχουν χρονική λειτουργικότητα), Aggregation και Transformation (κόμβοι οργάνωσης και ομαδοποίησης που υποστηρίζουν την ιεραρχία σε ένα γράφημα σκηνής), Geometry (κόμβοι αναπαράστασης γεωμετρικών σχημάτων), Geometric Properties (κόμβοι που ορίζουν τις βασικές ιδιότητες των γεωμετρικών κόμβων), Appearance (κόμβοι που περιγράφουν τις ιδιότητες εμφάνισης μιας γεωμετρίας και της σκηνής), Lighting (κόμβοι που φωτίζουν αντικείμενα στη σκηνή) και ένα σωρό άλλα χαρακτηριστικά που περιλαμβάνουν συστατικά όπως οι: Navigation (πλοήγηση), Interpolation, Text (κείμενο), Sound (ήχος), Pointing Device Sensor και Environmental Sensor (κόμβοι αισθητήρες), Texturing (υφή), Prototyping (προτυποποίηση) και Scripting [46][47].

Η αρχιτεκτονική X3D επιτρέπει, επίσης, διαστρωματώσεις από «προφίλ» που μπορούν να παρέχουν αυξημένη λειτουργικότητα για περιβάλλοντα εμπύθισης και μπορούν να υποστηριχθούν ανεξάρτητα. Συνεπώς, τα components μπορούν να επεκταθούν ή να τροποποιηθούν ανεξάρτητα, με την πρόσθεση νέων επιπέδων ή νέων συστατικών στοιχείων που μπορούν να προσφέρουν καινούρια χαρακτηριστικά, όπως το streaming. Το γεγονός αυτό καθιστά την X3D μια εύκολα επεκτάσιμη αρχιτεκτονική, όπου η εξέλιξη ενός συστατικού στοιχείου δεν καθυστερεί το σύνολο [48].

Τα βασικά προφίλ που παρέχει το X3D, ανάλογα με το σκοπό που θέλει να πετύχει ο προγραμματιστής και τις ιδιότητες που τον ενδιαφέρουν να χρησιμοποιήσει, είναι τα παρακάτω :

- **Interchange:** Είναι το βασικό προφίλ για την επικοινωνία μεταξύ των εφαρμογών. Υποστηρίζει γεωμετρικά σχήματα, υφές, βασικό φωτισμό και κίνηση. Δεν υπάρχει μοντέλο για rendering στο χρόνο εκτέλεσης, καθιστώντας το πολύ εύκολο να χρησιμοποιηθεί και να ενσωματωθεί σε οποιαδήποτε εφαρμογή.
- **Interactive:** Επιτρέπει βασική αλληλεπίδραση με το τρισδιάστατο περιβάλλον, προσθέτοντας διάφορους κόμβους-αισθητήρες που χρησιμοποιούνται για την περιήγηση και την αλληλεπίδραση του χρήστη με τον κόσμο (πχ. Plane Sensor, Touch Sensor κ.ά.), για τη βελτίωση του συγχρονισμού ή για επιπρόσθετο φωτισμό (πχ. Spot Light, Point Light).
- **Immersive:** Παρέχει τη δυνατότητα για πλήρη τρισδιάστατα γραφικά και πλήρη αλληλεπίδραση με τον χρήστη μέσω αισθητήρων, συμπεριλαμβάνοντας την ενσωμάτωση ήχου, την ανίχνευση συγκρούσεων 3D αντικειμένων, ομίχλης, και τη δυνατότητα scripting (μικρά κομμάτια κώδικα σε JavaScript) .
- **Full:** Συμπεριλαμβάνει όλους τους δηλωμένους κόμβους (nodes) που υπάρχουν στο πρότυπο, όπως επίσης και τις NURBS (Non-Uniform B-Splines) καμπύλες και επιφάνειες, το H-Anim (humanoid animation) πρότυπο και γεωχωρικά συστατικά (geospatial components).



Εικόνα 30: Πολυεπίπεδα προφίλ του X3D

Μερικά από τα πλεονεκτήματα της χρήσης του X3D είναι: α) το μικρό μέγεθος των αρχείων που το καθιστά ιδανικό για δικτυακές εφαρμογές, β) η υποστήριξη που προσφέρει για διαδικτυακές εφαρμογές, ώστε να μπορούν οι χρήστες να μοιράζονται αντικείμενα σε πραγματικό χρόνο, γ) η δυνατότητα λειτουργίας του σε όλες τις πλατφόρμες ανεξάρτητα από το λειτουργικό σύστημα που το φιλοξενεί, αφού είναι προσανατολισμένο στην εννοιολογική περιγραφή της σκηνής, δ) η δυναμικότητα που προσφέρει για την επέκτασή του σε συνδυασμό με την διαπεραστικότητα που μπορεί να έχει με τον χρήστη και ε) το γεγονός ότι είναι open-source αρχιτεκτονική και διατίθεται δωρεάν. Επίσης, μπορεί εκτός από την απεικόνιση τρισδιάστατων αντικειμένων να αναπαράγει και αρχεία βίντεο και ήχου [47].

Ο εικονικός κόσμος που δημιουργείται με το X3D δεν έχει τίποτα να ζηλέψει από αντίστοιχους κόσμους που δημιουργούνται με επαγγελματικά προγράμματα. Όπως διαπιστώσαμε παραπάνω, μπορούμε και εδώ να έχουμε κάμερες, αισθητήρες εξωτερικών συσκευών, αλλά και αισθητήρες κίνησης μέσα στον κόσμο. Με κατάλληλα scripts μπορούμε να εκμεταλλευτούμε τις εισόδους και εξόδους των αισθητήρων και να δημιουργήσουμε μια πολύ φιλική προς το χρήστη τρισδιάστατη εφαρμογή.

## 6.4 Γλώσσα Προγραμματισμού JAVA 3D

Η Java 3D είναι μια δυνατότητα που προσφέρει η Java για την αναπαράσταση τρισδιάστατων γραφικών. Πρόκειται για ένα γράφημα σκηνής βασισμένο σε ένα 3D API (Application Programming Interface) που τρέχει στην πλατφόρμα της Java. Τα προγράμματα που είναι γραμμένα σε Java 3D μπορούν να τρέχουν σε διαφορετικού είδους υπολογιστές, ακόμα και στο διαδίκτυο [49].

Η βιβλιοθήκη κλάσεων της Java 3D παρέχει μια πιο απλή διεπαφή από τις περισσότερες βιβλιοθήκες γραφικών και ενθουλακώνει τον προγραμματισμό γραφικών, χρησιμοποιώντας μια πραγματικά αντικειμενοστραφής προσέγγιση και προσφέροντας αρκετές δυνατότητες για να παράγει καλά παιχνίδια και animations. Μια σκηνή δομείται μέσω γραφήματος σκηνής, που αποτελεί μια αναπαράσταση των αντικειμένων που πρέπει να εμφανίζονται. Αυτό το γράφημα σκηνής έχει μια δένδροειδή ιεραρχία που εμπεριέχει αρκετά στοιχεία που είναι απαραίτητα για την εμφάνιση των αντικειμένων [49].

Η J3D χρησιμοποιεί προγραμματιστικό μοντέλο ενός DAG γραφήματος σκηνής, παρόμοια με της VRML και του X3D. Οφείλουμε, όμως, να σημειώσουμε πως τα γράφημα σκηνής της Java 3D είναι πιο δύσκολα να φτιαχτούν, δεδομένης της ίδιας της πολυπλοκότητας της Java ως γλώσσα, καθώς για κάθε αντικείμενο της σκηνής, κόμβο μετασχηματισμού ή συμπεριφορά, πρέπει να δημιουργείται ένα καινούριο στιγμιότυπο αντικειμένου, χρησιμοποιώντας τις αντίστοιχες κλάσεις της Java, να οριστούν και να αρχικοποιηθούν τα πεδία του αντικειμένου και έπειτα να προστεθούν τα αντικείμενα στη σκηνή [49].

Η Java 3D βασίζεται σε υπάρχουσα τεχνολογία, όπως η DirectX και η OpenGL, γεγονός που βοηθά σε ταχύτερη εκτέλεση των προγραμμάτων. Επίσης, η Java 3D μπορεί να ενσωματώσει αντικείμενα που έχουν δημιουργηθεί από άλλα πακέτα μοντελοποίησης, όπως αντικείμενα TrueSpace ή VRML.

Κάποια από τα πιο βασικά χαρακτηριστικά της είναι τα εξής:

- Πολυνηματική (Multithreaded) δομή γραφήματος σκηνής
- Ανεξαρτησία από το λειτουργικό σύστημα, καθώς τρέχει σε διαφορετικές πλατφόρμες
- Generic API πραγματικού χρόνου, που μπορούν να χρησιμοποιηθούν τόσο για απεικόνιση (visualization) όσο και για gaming
- Παροχή hardware-accelerated JOGL, OpenGL και Direct3D renderers (αναλόγως της πλατφόρμας)
- Παροχή εξελιγμένων μοντέλων εικονικής πραγματικότητας, βασισμένα στην όψη (viewmodels) και υποστήριξη στερεοσκοπικής απόδοσης και πολύπλοκων συνθέσεων multi-display.
- Εγγενής υποστήριξη για HDMs
- Υποστήριξη για CAVE (multiplescreen projectors)
- 3D χωρικό ήχο

Από την άλλη, η J3D δεν ορίζει κάποιο συγκεκριμένο μορφότυπο αρχείου, όπως άλλα προγραμματιστικά μοντέλα που χρησιμοποιούν γράφημα σκηνής. Ανταυτού, χρησιμοποιεί run-time loaders που φορτώνουν και υποστηρίζουν ένα εύρος 3D αρχείων, όπως VRML, X3D, Wavefront (OBJ), AutoCAD Drawing Interchange File (DXF), Caligari trueSpace (COB), Lightwave Scene Format (LSF), Lightwave Object Format (LOF), 3D-Studio (3DS).

## ΚΕΦΑΛΑΙΟ 7 - Εργαλεία Σχεδίασης Εικονικών Κόσμων

Προκειμένου να προχωρήσουμε στην ανάλυση απαιτήσεων και μετέπειτα στη σχεδίαση της εφαρμογής μας, οφείλουμε να δούμε, πέρα από τις υπάρχουσες δυνατότητες που προσφέρονται από άποψη προγραμματιστικών γλωσσών για την αναπαράσταση ενός εικονικού κόσμου, τι δυνατότητες προσφέρονται από άποψη εργαλείων σχεδίασης εικονικών κόσμων, εφόσον ένα τέτοιο εργαλείο σκοπεύουμε να φτιάξουμε. Ως εκ τούτου, μελετάμε τη δομή κάποιων τέτοιων εργαλείων και αναφέρουμε κάποιες από τις βασικές επιλογές που επιτρέπονται σε ορισμένα δημοφιλή εργαλεία σχεδίασης τρισδιάστατων αντικειμένων και εικονικών κόσμων, που χρησιμοποιούν γράφημα σκηνής, καθώς πιστεύουμε ότι μπορούν να εμπνεύσουν και την δική μας εφαρμογή.

### 7.1 Η Εφαρμογή Unity

Η εφαρμογή Unity είναι ένα λογισμικό ανάπτυξης εικονικών κόσμων και εικονικών πρακτόρων για τη δημιουργία βιντεοπαιχνιδιών. Έχει αναπτυχθεί με κώδικα C# και C++ και διαθέτει ένα πλούσιο σε δυνατότητες GUI που προσφέρει στο χρήστη μια ποικιλία έτοιμων αντικειμένων και εργαλείων για να αναπτύξει διαδραστικά μοντέλα σκηνής 2 ή 3 διαστάσεων. Διακρίνεται από την ποιότητα των υπηρεσιών που παρέχει όσον αφορά τα οπτικοακουστικά εφέ, ενώ διαθέτει παράλληλα και ένα προγραμματιστικό περιβάλλον στο χρήστη για την πρόσθεση scripts στη σκηνή καθώς και δυνατότητες testing. Επίσης, είναι ιδανικό για multi-player εφαρμογές, καθώς υποστηρίζει την ταυτόχρονη εκτέλεση σε διαφορετικές πλατφόρμες. Παρακάτω, παρουσιάζουμε κάποια από τα βασικά χαρακτηριστικά του προγράμματος, μελετώντας το interface του [50].

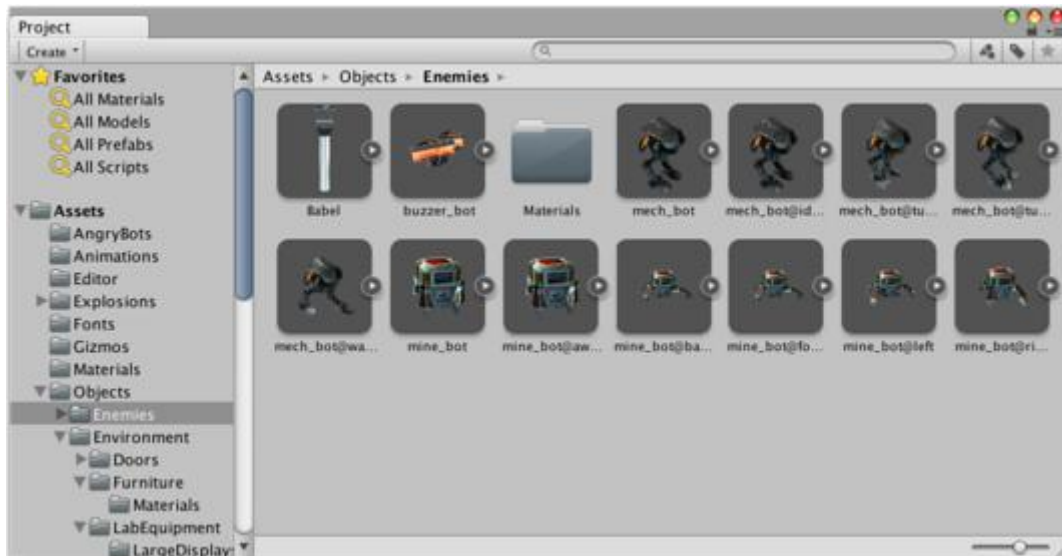
Αρχικά, το περιβάλλον ανάπτυξης διακρίνεται σε ένα μενού εργαλείων (Toolbar), όπου προσφέρονται οι βασικές λειτουργίες που χρησιμοποιούνται ευρέως στο πρόγραμμα και τέσσερα βασικά παράθυρα (views) που συνυπάρχουν και παρουσιάζουν διαφορετικά χαρακτηριστικά της εφαρμογής και είναι τα: Scene, Inspector, Hierarchy και Project [50].



Εικόνα 31: Το γραφικό περιβάλλον χρήστη του Unity

Το Project view επιτρέπει την πρόσβαση στα αντικείμενα που ανήκουν στη συγκεκριμένη εργασία, εμφανίζοντας τη δομή των φακέλων που αποτελούν το project σαν μια λίστα ιεραρχίας. Επιλέγοντας ένα φάκελο, εμφανίζονται δεξιά τα χαρακτηριστικά του μαζί με εικόνες μικρογραφίας των αντικειμένων που εμπεριέχει ο φάκελος. Το παράθυρο προσφέρει και μια γραμμή αναζήτησης, όπου ο χρήστης μπορεί να πληκτρολογήσει το όνομα ενός

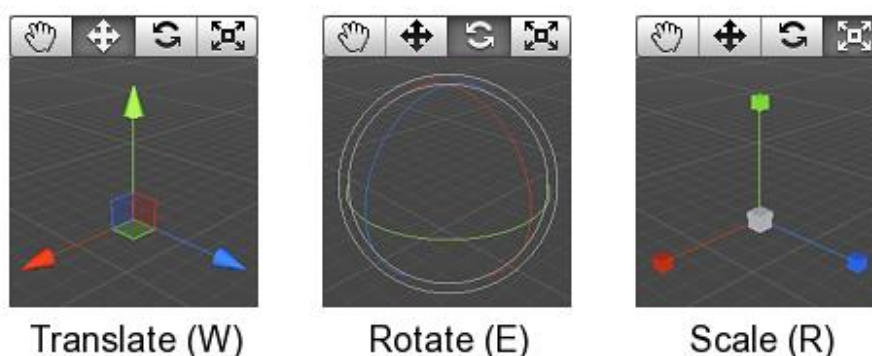
αντικείμενου που θέλει να αναζητήσει για να διαχειριστεί. Επίσης, ο χρήστης μπορεί να σύρει (draganddrop) ένα αντικείμενο από το Project view στο Hierarchy ή Scene View, ώστε αυτό να εμφανιστεί τη σκηνή [50].



Εικόνα 32: Project tab

Το Hierarchy περιέχει κάθε αντικείμενο που εμφανίζεται στην τρέχουσα σκηνή σε μορφή ιεραρχίας, ανάλογα τη σύνδεση που υπάρχει μεταξύ τους. Ο χρήστης μπορεί να επιλέξει ένα αντικείμενο και να το σύρει σε κάποιο άλλο, μετατρέποντάς το σε κόμβο-παιδί αυτού του αντικειμένου. Ο κόμβος-παιδί κληρονομεί κάθε κίνηση και περιστροφή που γίνεται στον κόμβο-γονέα [50].

Το Scene view είναι το περιβάλλον όπου απεικονίζεται η σκηνή και τοποθετούνται τα αντικείμενα. Για την επιλογή των αντικειμένων και την πλοήγηση στη σκηνή χρησιμοποιούνται επιλογές από το Toolbar ή συνδυαστικά το ποντίκι με κάποιο keystroke (από το πληκτρολόγιο). Από το Toolbar δίνεται η δυνατότητα Transform που επιτρέπει την μετακίνηση, την περιστροφή και την μεγέθυνση/σμίκρυνση ενός αντικειμένου. Επίσης, δίνεται η δυνατότητα περιστροφής και μετατόπισης της κάμερας λήψης πάνω στους άξονες xyz, αλλάζοντας την οπτική γωνία παρατήρησης της σκηνής [50].

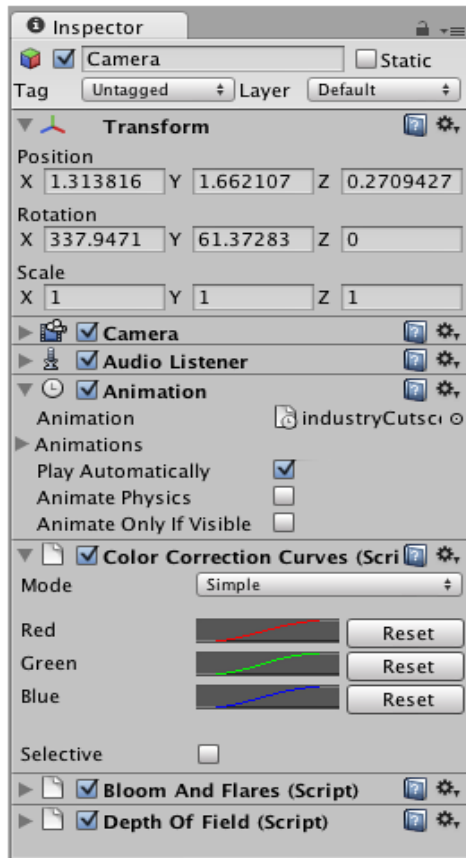


Εικόνα 33: Transform επιλογές

Το Inspector view προβάλλει λεπτομερείς πληροφορίες για το αντικείμενο της σκηνής που διαλέγει ο χρήστης, συμπεριλαμβανομένου των ιδιοτήτων του και των συστατικών από τα οποία μπορεί να συντίθεται. Σε αυτό το παράθυρο, ο χρήστης μπορεί να μεταβάλλει τη

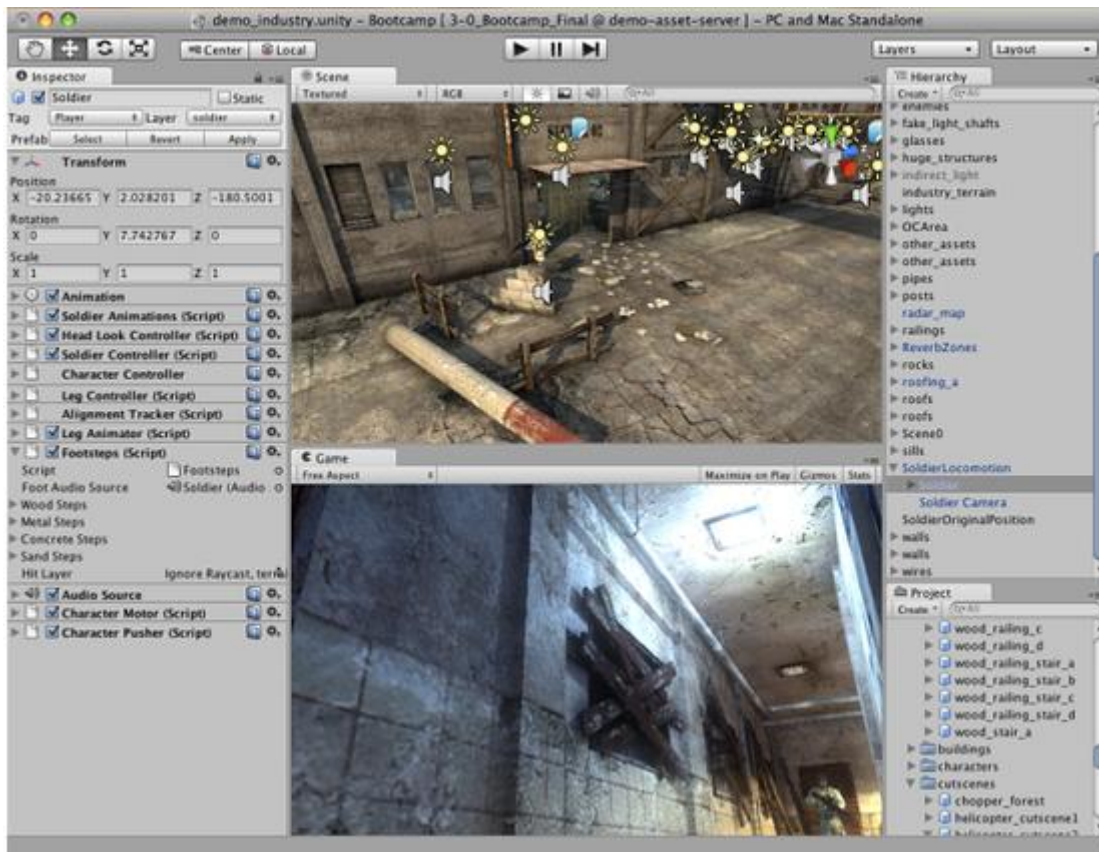


λειτουργικότητα αντικειμένων στην σκηνή, προσθέτοντας συστατικά όπως είναι ο φωτισμός ή τα Scripts, αλλάζοντας παραμέτρους και ελέγχοντας τη συμπεριφορά κατά την εκτέλεση [50].



Εικόνα 34: Inspector view

Τέλος, ο χρήστης μπορεί να προσαρμόσει το περιβάλλον εργασίας του με βάση τις ανάγκες που έχει, εμφανίζοντας κάθε view σε διαφορετικό παράθυρο ή προσαρμόζοντας την οθόνη με εναλλακτικό τρόπο και σώζοντας αυτές τις αλλαγές ως νέο layout, όπως φαίνεται παρακάτω:



Εικόνα 35: Μεταβολή των γωνιών θέασης στο περιβάλλον της εφαρμογής

## 7.2 Η Εφαρμογή 3DStudioMax

Μια επίσης δημοφιλής εφαρμογή για την παραγωγή τρισδιάστατων γραφικών σε υπολογιστή, με τη δημιουργία τρισδιάστατων μοντέλων, εικόνων, animations και βιντεοπαιχνιδιών είναι το πρόγραμμα της Autodesk, 3DStudioMax. Χρησιμοποιείται κατά κόρον για αρχιτεκτονικό σχεδιασμό, ανάπτυξη εικονικών κόσμων για βιντεοπαιχνίδια και ταινίες animation και υποστηρίζεται από το λειτουργικό σύστημα των Windows. Παρότι δεν προσφέρεται δωρεάν και υπάρχει μια απόκλιση από τις δυνατότητες που μας ενδιαφέρουν και τις λειτουργίες που προορίζεται, αξίζει να δούμε κάποιες από τις βασικές του λειτουργίες, μελετώντας το interface του [51].

Το περιβάλλον ανάπτυξης διακρίνεται στα παρακάτω:

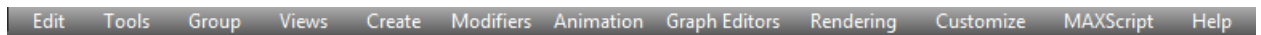
- **QuickAccessToolbar**: Ένα μενού εργαλείων στην πρώτη γραμμή του περιβάλλοντος ανάπτυξης που παρέχει τις πιο βασικές εντολές που αφορούν τη διαχείριση αρχείων (άνοιγμα αρχείου, άνοιγμα φακέλου, αποθήκευση), εντολές αναίρεσης ή επανεκτέλεσης καθώς και μια drop-down λίστα για εναλλαγή μεταξύ των διαφορετικών παραθύρων εργασίας.



Εικόνα 36: Quick Access Toolbar

- **Menu**: Το μενού με σύνοψη όλων των διαθέσιμων επιλογών που προσφέρει η εφαρμογή, ανά κατηγορία ενέργειας.





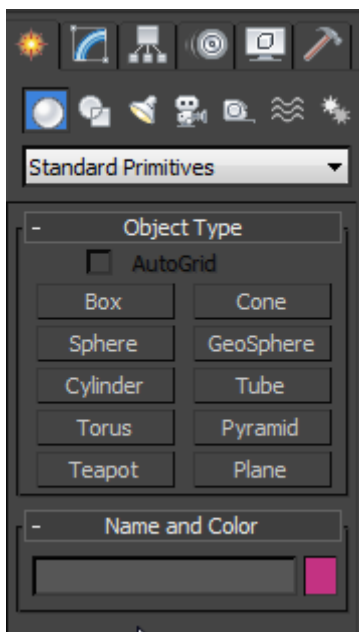
Εικόνα 37: Κεντρικό Menu

- **MainToolbar**: Πρόκειται για ένα μενού εργαλείων που παρέχει γρήγορη πρόσβαση σε εργαλεία και παράθυρα που χρησιμοποιούνται ευρέως για τις πιο βασικές ενέργειες που εκτελούνται στο πρόγραμμα.



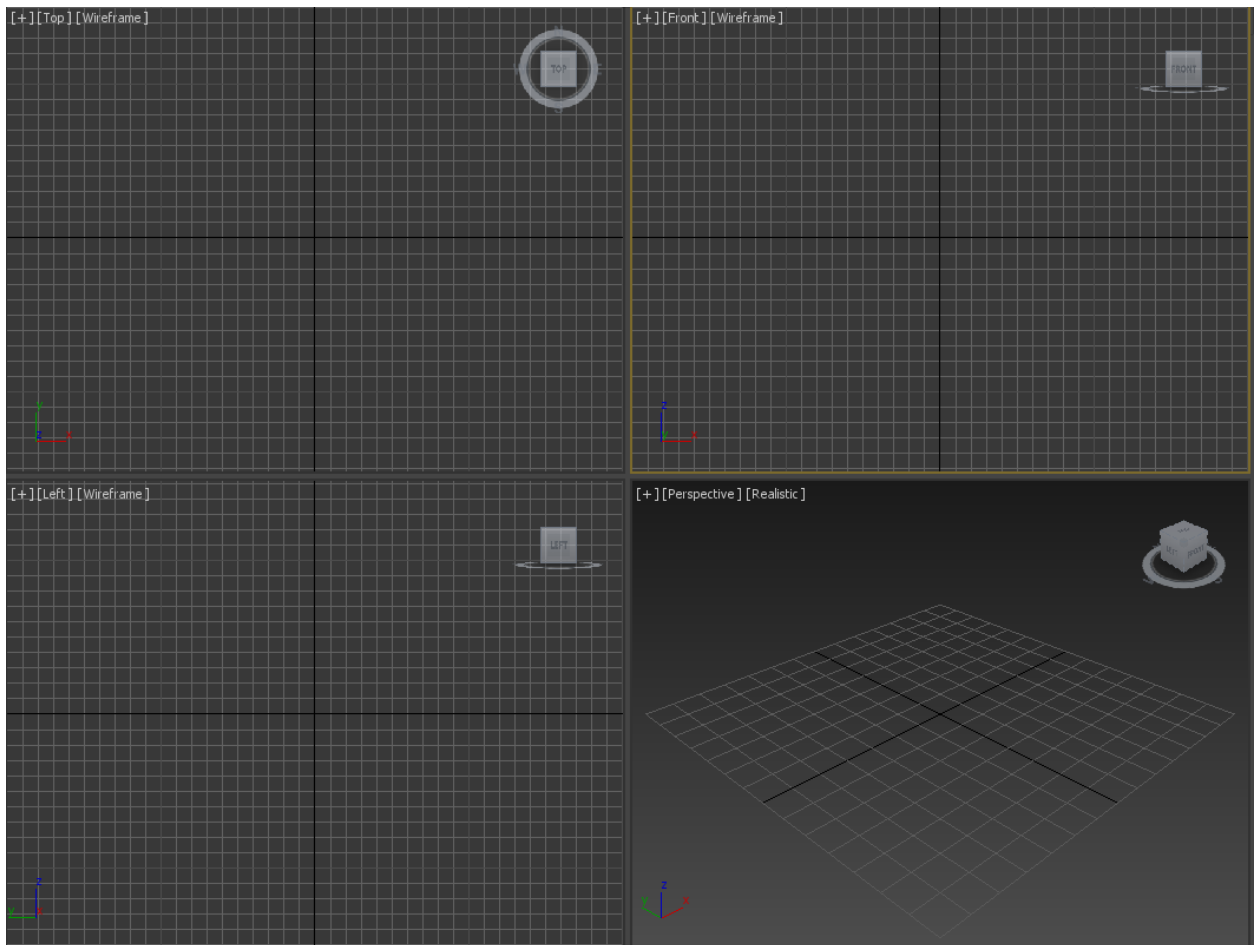
Εικόνα 38: Κεντρική εργαλειοθήκη

- **GraphiteModelingTools**: Πρόκειται για μια περιοχή εργαλείων στο πάνω μέρος της οθόνης που χρησιμοποιείται για τη διευκόλυνση της σχεδίασης αντικειμένων (μοντελοποίηση, χρωματισμός σκηνής).
- **CommandPanel**: Αποτελείται από έξι UI panels, στα οποία μπορεί να πλοηγηθεί εναλλάξ ο χρήστης. Αυτά δίνουν πρόσβαση σε εργαλεία που αφορούν τη δημιουργία ή την επεξεργασία της γεωμετρίας των αντικειμένων, την προσθήκη πολυπλοκότητας σε μια γεωμετρία, την προσθαφαίρεση φωτισμού, τον έλεγχο των animations, που αφορούν την μοντελοποίηση, τα animations, επιλογές εμφάνισης μιας σκηνής ή ενός αντικειμένου και άλλες δυνατότητες.



Εικόνα 39: Command Panel

- **Viewport**: Πρόκειται για 4 διαφορετικά παράθυρα όπου εμφανίζεται η σκηνή υπό διαφορετικές οπτικές γωνίες. Επιπλέον, ο χρήστης μπορεί να επιλέξει εργαλεία για να ρυθμίσει κάποιο από τα παράθυρα να εμφανίζει με διαφορετικό τρόπο τη σκηνή, πχ wireframe ή shaded.



Εικόνα 40: Viewports

- **Viewport Label Menus:** Είναι μενού που επιτρέπει στο χρήστη να αλλάξει τι εμφανίζει κάθε viewport, συμπεριλαμβανομένου την οπτική γωνία παρατήρησης και το στυλ της σκίασης.
- **Viewport Navigation controls:** Κουμπιά που επιτρέπουν την πλοήγηση στη σκηνή στο τρέχον viewport.



Εικόνα 41: Επιλογές πλοήγησης

- **Slate Material Editor:** Προσφέρει ρυθμίσεις για την επεξεργασία των υλικών και των παρς των αντικειμένων, ώστε να επιτυγχάνεται μεγαλύτερος ρεαλισμός.
- **Time Slider:** Αφορά τα animations και επιτρέπει την χρονική πλοήγηση σε αυτά και την μετάβαση σε διαφορετικό frame στο πέρασμα του χρόνου.



Εικόνα 42: Time Slider

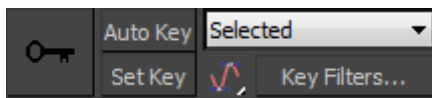
- **Status Bar Controls:** Πρόκειται για μια περιοχή στο τέλος του παραθύρου όπου παρέχονται πληροφορίες για την κατάσταση της σκηνής. Επίσης, στα δεξιά

εμφανίζονται στοιχεία για τις συντεταγμένες, όπου μπορεί χειροκίνητα ο χρήστης να μεταβάλλει τις τιμές για να μετατοπίσει ένα αντικείμενο. Στα αριστερά, προσφέρεται η δυνατότητα εισαγωγής μιας γραμμής script βάσει του MAXScriptListener που χρησιμοποιεί η εφαρμογή. Η MAXScript είναι μια built-in scripting γλώσσα που χρησιμοποιείται για να αυτοματοποιήσει ο χρήστης επαναλαμβανόμενες εργασίες, για να συνδυάσει υπάρχουσες λειτουργικότητες για διαφορετικό σκοπό, για να αναπτύξει νέα εργαλεία και interfaces.



Εικόνα 43: Status Bar Controls

- **Animation & Time Controls:** Πρόκειται για επιλογές που μπορούν να μεταβάλλουν ένα animation στη διάρκεια του χρόνου.



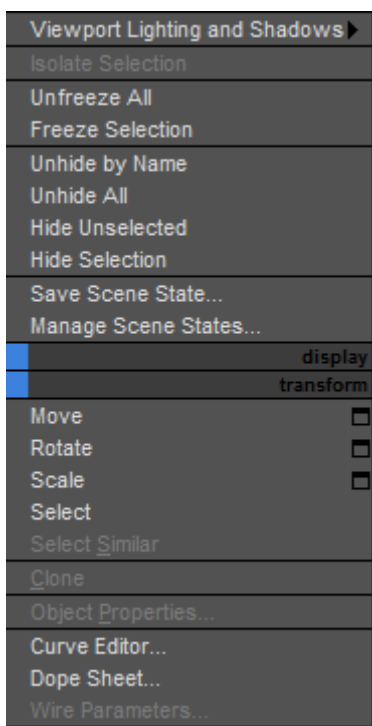
Εικόνα 44: Animation & Time controls

- **Animation Play back Controls:** Κουμπιά που επιτρέπουν στο χρήστη να δει την σκηνή σε κίνηση στη διάρκεια του χρόνου.



Εικόνα 45: Playback controls

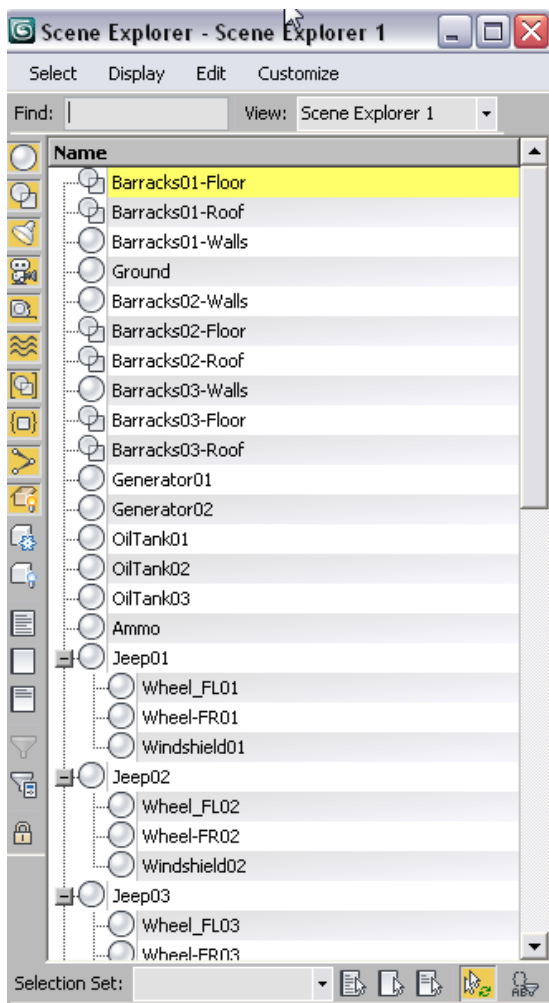
- **Rendered Frame Window:** Σε αυτό το παράθυρο γίνεται rendering της σκηνής με τις κατάλληλες ρυθμίσεις που θα ορίσει ο χρήστης.
- **Quad Menu:** Πρόκειται για ένα μενού που εμφανίζεται όταν ο χρήστης πατάει δεξί κλικ σε μια περιοχή του viewport. Οι επιλογές που εμπεριέχει εξαρτώνται από το αντικείμενο που επιλέγεται.



Εικόνα 46: Quad Menu σε άδεια σκηνή

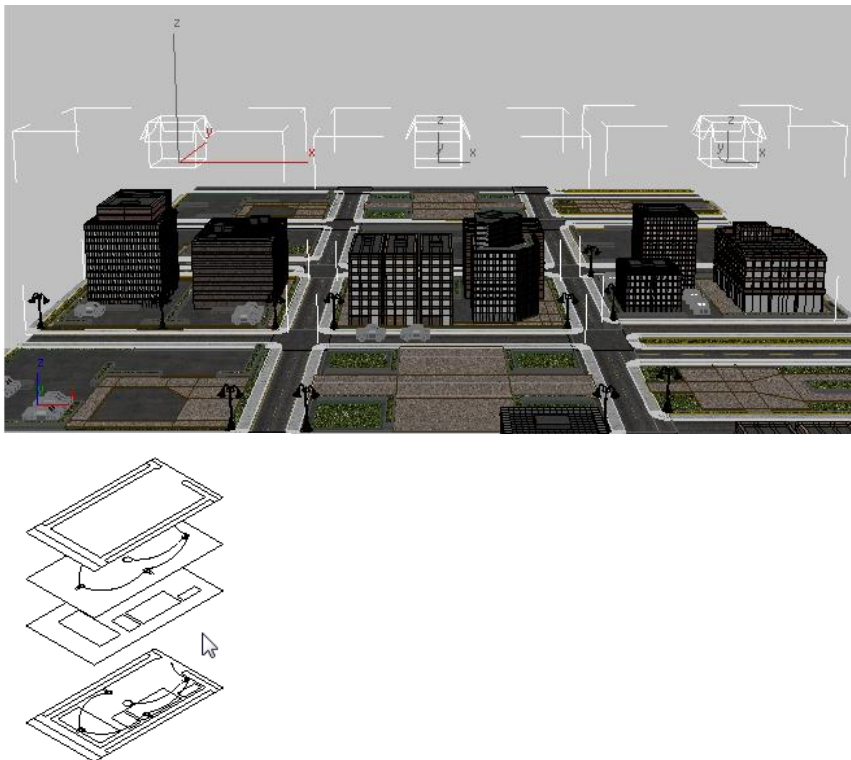
Για τη διαχείριση των σκηνών, των αρχείων και των φακέλων/projects προσφέρονται τα παρακάτω εργαλεία [51]:

- **Scene Explorer:** Είναι ένα δυνατό εργαλείο για να βλέπει ο χρήστης τις ιδιότητες του αντικειμένου που μπορούν να μεταβληθούν, για να επιλέγει αντικείμενα βάσει διάφορων κριτηρίων και για να δημιουργεί και να τροποποιεί ιεραρχίες αντικειμένων.



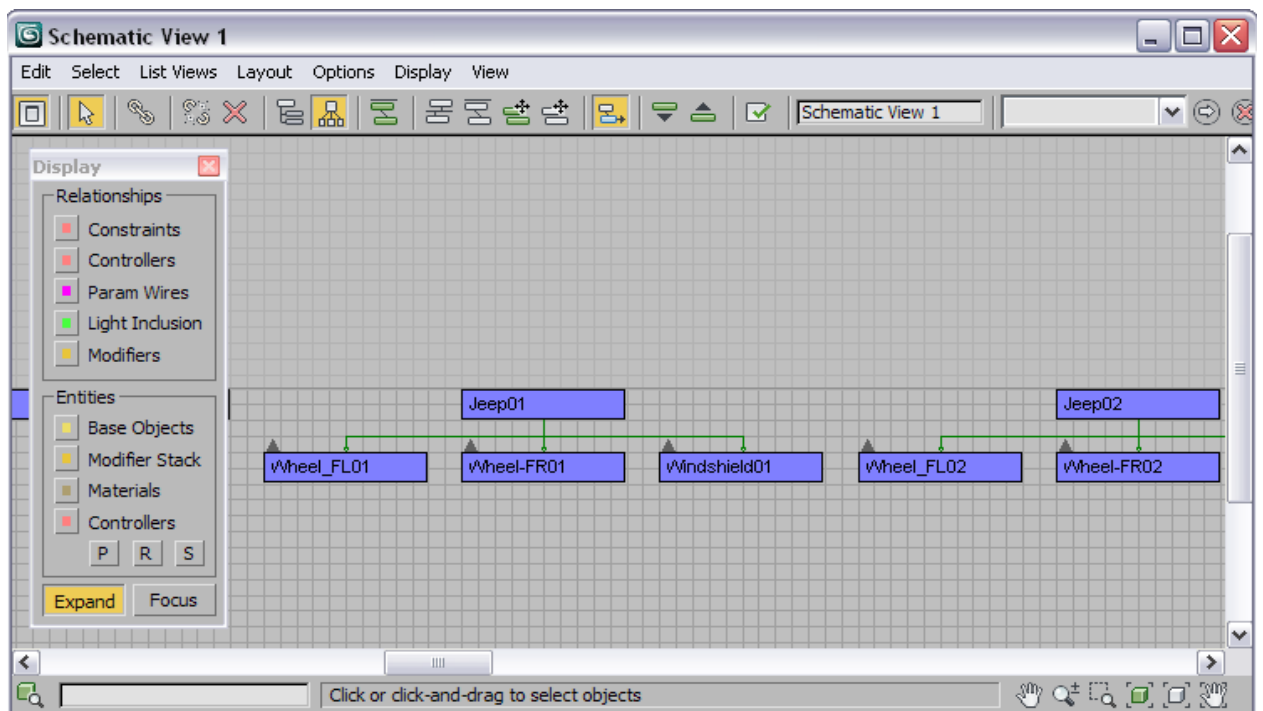
Εικόνα 47: Scene Explorer

- Organizational Tools:** Εδώ υπάγονται τα Container, Group και Layer. Τα δυο πρώτα χρησιμεύουν για να οργανώνουν τα αντικείμενα βάσει της λογικής τους συσχέτισης, έτσι ώστε μια τέτοια ομάδα αντικειμένων να μπορεί να τη χειριστεί ο χρήστης ως ένα αντικείμενο. Το Group προσφέρει προσθήκη βασικής λειτουργικότητας, ενώ το Container παρέχει προχωρημένες ιδιότητες, όπως η προσβασιμότητα και η κοινή χρήση αρχείων και κανόνες κληρονομικότητας. Το Layer είναι σαν μια διάφανη επιστρωση πάνω στην οποία ο χρήστης μπορεί να οργανώσει και να ομαδοποιήσει διαφορετικά είδη πληροφορίας για τη σκηνή. Αντικείμενα στο ίδιο layer μπορούν να μοιράζονται ίδιο χρώμα, ρυθμίσεις rendering και ρυθμίσεις εμφάνισης [51].



Εικόνα 48: Container (αριστερά) και Layer (δεξιά)

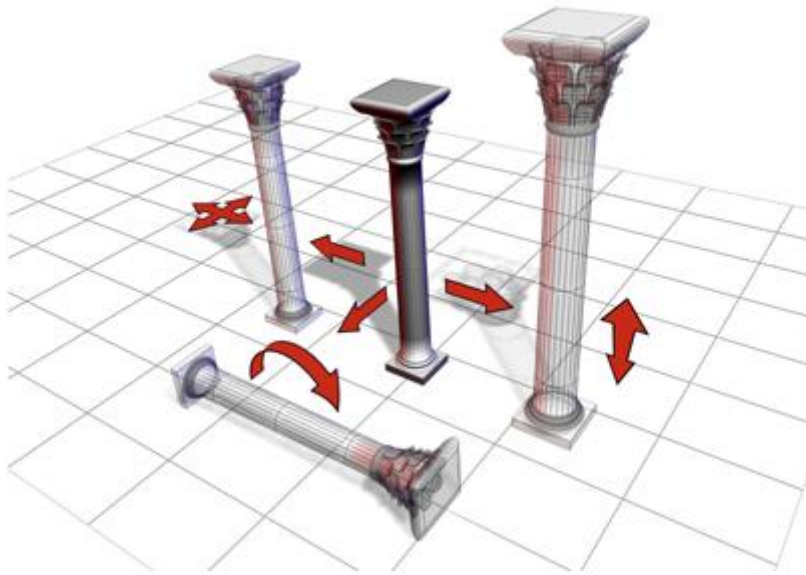
- **State Sets και Scene Sets:** Προσφέρουν γρήγορους τρόπους αποθήκευσης διαφορετικών καταστάσεων της σκηνής με διαφορετικές ιδιότητες στις οποίες μπορεί να ανατρέξει ο χρήστης όποτε θέλει, για να παράγει διαφορετικές ερμηνείες rendering ενός μοντέλου.
- **Schematic View:** Εμφανίζει τη σκηνή σαν ένα γραφικό σχήμα, αντί για γεωμετρία. Προσφέρει έναν εναλλακτικό τρόπο στο χρήστη να επιλέξει ή να μετονομάσει αντικείμενα στη σκηνή και να πλοηγηθεί μεταξύ των τροποποιήσεων (modifiers). Είναι εξαιρετικά χρήσιμο για την ιεραρχική απεικόνιση των αντικειμένων.



Εικόνα 49: Schematic View

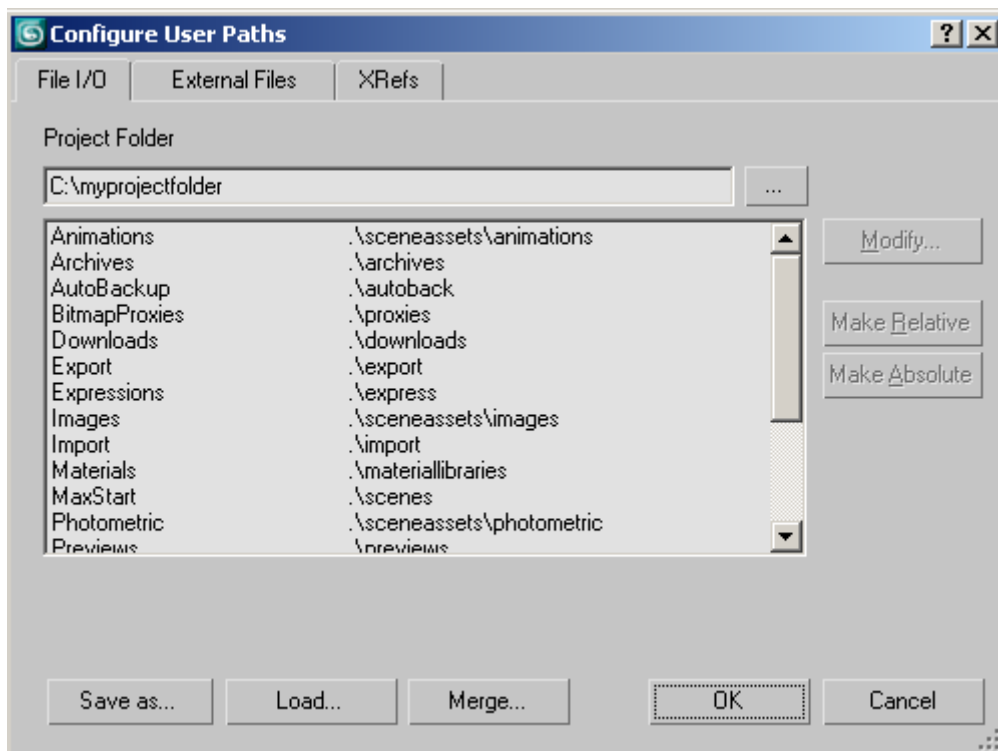
Στο 3DStudioMax λοιπόν, ο χρήστης μπορεί να σχεδιάσει αντικείμενα τρισδιάστατης γεωμετρίας ή δισδιάστατα σχήματα και στη συνέχεια να τα επεξεργαστεί, μέσω των σχετικών εργαλείων που προσφέρονται. Επίσης, το πρόγραμμα προσφέρει ένα σύνολο έτοιμων αντικειμένων για χρήση. Οι αλλαγές που υφίσταται ένα αντικείμενο αποθηκεύονται σε μια στοίβα, δίνοντας τη δυνατότητα στο χρήστη να αλλάξει την επίδραση που είχε μια τροποποίηση, αφαιρώντας την προκείμενη επεξεργασία από τη στοίβα.

Η επιλογή ενός αντικειμένου είναι ουσιαστική για την εκτέλεση κάποιας ενέργειας επί αυτού και γίνεται με διάφορους τρόπους. Οι πιο κοινά είναι με το ποντίκι ή το πληκτρολόγιο ή σύροντας μια περιοχή τριγύρω από αυτό. Ακόμα, η επιλογή μπορεί να γίνει βάση του ονόματος του αντικειμένου ή άλλων ιδιοτήτων του όπως το χρώμα ή η κατηγορία στην οποία υπάγεται. Επιπλέον, η επιλογή μπορεί να γίνει μέσω ενός schematic view, δηλαδή ενός παραθύρου που εμφανίζει τα αντικείμενα της τρέχουσας σκηνής σε ιεραρχία. Τέλος, μετά την επιλογή μπορεί να γίνει η τοποθέτηση των αντικειμένων στην κατάλληλη θέση, χρησιμοποιώντας εργαλεία transform για μετατόπιση (Move), για περιστροφή (Rotate) και για κλιμάκωση (Scale) και να γίνει ευθυγράμμιση (Align) των αντικειμένων βάσει ενός άξονα. Ακόμη, ο χρήστης μπορεί να κλειδώσει ή να κρύψει κάποια αντικείμενα ή μια κατηγορία αντικειμένων για να αποφύγει μη-επιθυμητή τροποποίησή τους, ενώ μπορεί μετά να τα ξεκλειδώσει ή να τα επανεμφανίσει [51].



Εικόνα 50: Παράδειγμα κολώνας σε μετατόπιση, κλιμάκωση και περιστροφή

Η εφαρμογή επιτρέπει την εισαγωγή (Import) και επεξεργασία πλήθους τύπων αρχείου, πέρα από τα .3ds αρχεία που παράγει το ίδιο το πρόγραμμα. Επίσης, όταν επιλέγεται το άνοιγμα ενός αρχείου, η εφαρμογή θυμάται το τελευταίο μονοπάτι που χρησιμοποιήθηκε προς διευκόλυνση του χρήστη. Ωστόσο, προσφέρει και δυνατότητες ρύθμισης των μονοπατιών που χρησιμοποιούνται για διαφορετικά είδη αρχείων [51].

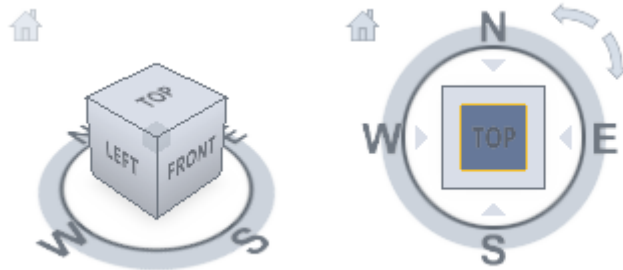
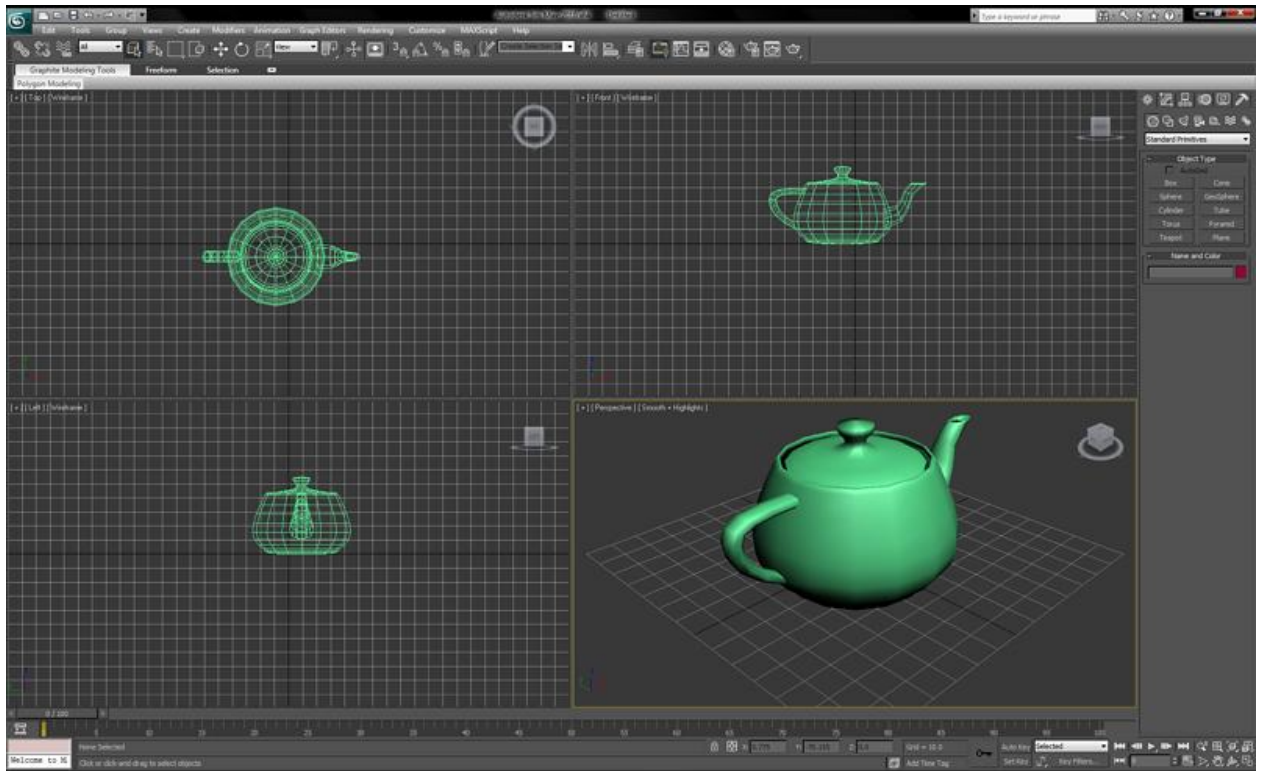


Εικόνα 51: Ρύθμιση μονοπατιών χρήστη

Κάθε αντικείμενο στη σκηνή τοποθετείται σε ένα τριδιάστατο κόσμο, τον οποίο ο χρήστης παρακολουθεί μέσα από ένα ή παραπάνω viewports. Υπάρχει μια ποικιλία επιλογών για την οπτικοποίηση αυτού το χώρου, από την πλήρη απλούστευση ως την εξαιρετική

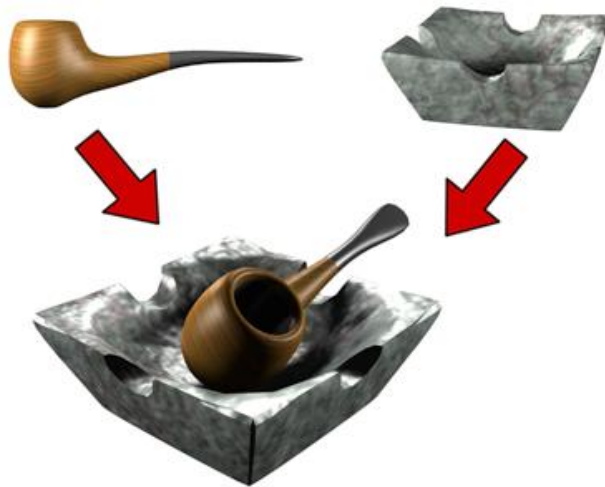


λεπτομέρεια που μπορεί να εμφανίσει. Κάθε viewport μπορεί να εμφανίσει δυο είδη όψεων: την αξονομετρική ή την προοπτική. Η περιήγηση στον τριδιάστατο χώρο γίνεται με την προσαρμογή της θέσης, της περιστροφής και του μεγέθους των views. Επίσης, ο προσανατολισμός ενός view μπορεί να αλλάξει με τη βοήθεια του εργαλείου View Cube, που εμφανίζεται σε κάθε view και είναι ένας τρισδιάστατος κύβος που μας φανερώνει ποια πλευρά του viewport βλέπουμε και ποιος ο προσανατολισμός του.



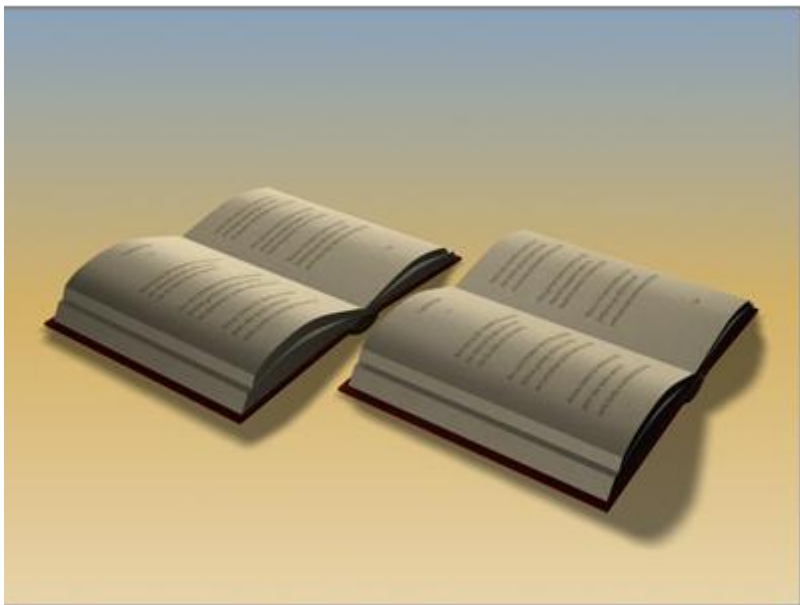
Εικόνα 52: Teapot σε view ports και σε Cube View

Επίσης, μια ενδιαφέρουσα επιλογή είναι το Merge όπου διαφορετικές σκηνές συνδυάζονται για την παραγωγή μιας νέας σκηνής. Κατά την σύντηξη ενός αρχείου, ο χρήστης μπορεί να επιλέξει ποια αντικείμενα να συγχωνεύσει. Ακόμη, προσφέρεται η δυνατότητα της αντικατάστασης (Replace) ενός αντικειμένου με ένα άλλο, με τη προϋπόθεση ότι έχουν το ίδιο όνομα. Αυτό γίνεται για απλοποίηση της σκηνής, χρησιμοποιώντας πιο ελαφριά αντικείμενα και αντικαθιστώντας τα με τα τελικά μοντέλα πριν το rendering [51].



**Εικόνα 53: Συγχώνευση αντικειμένων**

Τέλος, ο χρήστης μπορεί να δημιουργήσει διαφορετικά είδη αντιγράφων ενός αντικειμένου: copies, instances ή references. Το copy δημιουργεί ένα πιστό αντίγραφο του αντικειμένου, αλλαγές στο οποίο δεν επηρεάζουν το αρχικό. Αντίθετα, το instance είναι αντίγραφο του πρωτότυπου, αλλαγές στο οποίο αντικατοπτρίζονται και στο πρωτότυπο. Το reference, από την άλλη, είναι ένας συνδυασμός των δυο, με την έννοια ότι αλλαγές στις παραμέτρους που είχε και το πρωτότυπο όταν δημιουργήθηκε το αντίγραφο, θα επηρεάσει και τα δυο αντικείμενα, ενώ τροποποιήσεις σε άλλες παραμέτρους επιδρούν μόνο στο αντίγραφο [51].



**Εικόνα 54: Δημιουργία αντιγράφου**

Οι ρυθμίσεις και οι επιλογές που προσφέρει η εφαρμογή 3DStudioMax είναι πραγματικά πολλές και καλύπτουν ένα εύρος περιπτώσεων, καθιστώντας το πρόγραμμα δύσκολο στον χειρισμό για έναν αρχάριο, αλλά παράλληλα προσφέροντας φοβερές δυνατότητες για τη δημιουργία ρεαλιστικών σκηνών. Σε αυτό το κεφάλαιο, κάναμε μια σύντομη αναφορά στη λειτουργικότητα του, μελετώντας το user interface και εστιάζοντας μονάχα σε ένα μικρό υποσύνολο των εργαλείων του, βάσει των αναγκών που θέλουμε να εξυπηρετήσει και η δική μας εφαρμογή [51].

## Κεφάλαιο 8 - Η Πλατφόρμα Reve

Στο κεφάλαιο αυτό θα γίνει μια σύντομη αναφορά στην πλατφόρμα REVE, η οποία αποτελεί μια αφηρημένη αναπαράσταση εικονικών κόσμων που αναπτύχθηκε από τον Γ. Αναστασάκη και στα πλαίσια της οποίας πραγματοποιείται η συγκεκριμένη διπλωματική διατριβή. Η πλατφόρμα REVE είναι μια ολοκληρωμένη και πλήρως υλοποιημένη και λειτουργική πλατφόρμα ανάπτυξης λογισμικού. Χρησιμοποιείται για την ανάπτυξη και την λειτουργία εικονικών περιβαλλόντων. Βασικά συστατικά της είναι μία παραδειγματική υλοποίηση της αναπαράστασης αντικειμένων και εικονικών σωμάτων αλλά και η διασύνδεση της με άλλα τμήματα καθώς και την παρακολούθηση της λειτουργίας της για σκοπούς διερεύνησης, βελτιστοποίησης και αποσφαλμάτωσης. Η πλατφόρμα είναι υλοποιημένη στη γλώσσα Java [52].

### 8.1 Τι Είναι Η Πλατφόρμα REVE

Η πλατφόρμα REVE αποτελεί μία αφηρημένη αναπαράσταση για εικονικά περιβάλλοντα η οποία στοχεύει στην τυποποιημένη ανάπτυξη εικονικών περιβαλλόντων ενώ προσφέρει ευελιξία, επεκτασιμότητα καθώς και ρεαλιστικό σχεδιασμό. Το εικονικό περιβάλλον είναι ένα σύνολο εικονικών αντικειμένων τα οποία περιέχουν το σύνολο της διαθέσιμης πληροφορίας του κόσμου. Καταλαβαίνουμε ότι, οτιδήποτε θέλουμε να ενσωματώσουμε στα πλαίσια του κόσμου μας θα πρέπει να αντιστοιχεί σε κάποιο αντικείμενο. Για αυτό το λόγο κάθε αντικείμενο αναπαριστάται από μια δομή δεδομένων η οποία καλείται item [52].

Στον πραγματικό κόσμο τα φυσικά αντικείμενα τα οποία αντιλαμβάνεται ο άνθρωπος διέπονται από ένα σύνολο χαρακτηριστικών τόσο φυσικών όσο και λειτουργικών. Έτσι και στην δομή item θα πρέπει να περιέχονται όλα τα χαρακτηριστικά του αντικείμενου τόσο από φυσικής όσο και από λειτουργικής άποψης. Πιο συγκεκριμένα θα πρέπει να περιέχονται πληροφορίες για το μέγεθος, το προσανατολισμό, την τοποθεσία στην οποία βρίσκεται το item. Γενικά θα πρέπει να δίνονται πληροφορίες για την φυσική αναπαράσταση (σχήμα, μέγεθος, χρώμα, υφή, υλικό, θέση) αλλά και πληροφορίες σχετικά με την λειτουργικότητα του όπως είναι η δυνατότητα μετακίνησης, περιστροφής και γενικότερα κάθε αλλαγής ως προς την φυσική αναπαράσταση του item (αναπαράσταση λειτουργικότητας). Επιπρόσθετα οι πληροφορίες θα πρέπει να είναι διαθέσιμες ως προς κάθε δέκτη με ένα κατάλληλο επίπεδο αφαίρεσης έτσι ώστε ο κάθε δέκτης να μπορεί να χρησιμοποιήσει την πληροφορία με τον τρόπο που θέλει καθώς και στο εύρος που θέλει (σημασιολογική αναπαράσταση) [52].

Στην REVE προκειμένου να καλυφθούν οι παραπάνω κατηγορίες αναπαράστασης εισάγονται τρία item aspects ως επιμέρους δομές ενός item. Τρεις δηλαδή «απόψεις» αναπαράστασης της πληροφορίας, οι οποίες είναι οι ακόλουθες:

- **Physical aspect** : Η οποία αντιστοιχεί στην φυσική αναπαράσταση και σχετίζεται με τα «φυσικά» χαρακτηριστικά του εικονικού item (μέγεθος, σχήμα)
- **Semantic aspect** : Η οποία αντιστοιχεί στην σημασιολογική αναπαράσταση και σχετίζεται με την πληροφορία την οποία μπορούν να δεχθούν οι δέκτες (εδώ εικονικοί πράκτορες)
- **Virtual aspect** : Η οποία αντιστοιχεί στην λειτουργική αναπαράσταση και σχετίζεται με την λειτουργικότητα του item

Η πλατφόρμα REVE διαθέτει προηγμένες δυνατότητες οπτικοποίησης εικονικών κόσμων, υποστήριξης πολλαπλών εικονικών πρακτόρων, κατανεμημένης λειτουργίας, επέκτασης και ανάπτυξης εφαρμογών.

Βασίζεται στο πρωτόκολλο AIP, το οποίο χρησιμοποιείται για τη σύνδεση εικονικών σωμάτων με συστατικά παραγωγής συμπεριφοράς και γνωστικής λειτουργίας (cognition), υποστηρίζοντας έτσι ποικιλία μέσων ελέγχου των ορισμών και των σωμάτων. Επιτρέπει τη σύνδεση στην πλατφόρμα εξωτερικών εφαρμογών παραγωγής συμπεριφοράς μέσω πρωτοκόλλου TCP/IP και καθιστά δυνατή την παρακολούθηση της εκτέλεσης σε πραγματικό χρόνο [52].

Για τον ορισμό εικονικών κόσμων, ως συνόλων εικονικών αντικειμένων χρησιμοποιείται η βασιζόμενη σε XML γλώσσα VERL (Virtual Environment Representation Language). Η γλώσσα VERL (Virtual Environment Representation Language) αναπτύχθηκε το 2010 από τον Γ. Αναστασάκη, αναπληρωτή καθηγητή του Πανεπιστημίου Πειραιώς, για την αναπαράσταση εικονικών κόσμων. Η γλώσσα VERL χρησιμοποιεί δεδομένα X3D σε κωδικοποίηση VRML και εφαρμόζει σε αυτά χωρικούς μετασχηματισμούς [52].

## Κεφάλαιο 9 - Η Γλώσσα VERL

Ένα πολύ σημαντικό κομμάτι της αναπαράστασης REVE είναι η γλώσσα VERL ( Virtual Environment Representation Language ) μια απλή γλώσσα βασισμένη στην μεταγλώσσα XML η οποία είναι ειδικά σχεδιασμένη για την αναπαράσταση εικονικών περιβαλλόντων και ακολουθεί τις αρχές αναπαράστασης της REVE.

### 9.1 Τι Είναι Η Γλώσσα VERL

Στην γλώσσα VERL ορίζεται ο εικονικός κόσμος (world) ο οποίος αποτελείται από items τα οποία ανήκουν σε συγκεκριμένα itemGroups. Κάθε ένα από αυτά τα itemGroups ή τα items μπορούν να αποδοθούν συγκεκριμένα χαρακτηριστικά όπως π.χ. class: real/unreal, behaviors: true/false. Επιπρόσθετα αναπαριστούνται τα τρία item aspects (Physical aspect, Semantic aspect, Virtual aspect), που αναφέραμε σε παραπάνω κεφάλαιο [52].

Μια αντιπροσωπευτική αναπαράσταση ενός αντικειμένου είναι η ακόλουθη :

```
<itemGroup name="example" location="example.wrl">
  <item name="example1">
    <virtualModel source="moreexample">
      <transform translation="0, 0, 0" rotation="0 0 0 1" />
    </virtualModel>
    <accessModel>
      <accessPoint name="example_access" node="top">
        <function name="rotate"
          class="reve.scene.item.access.function.X3DFieldFunctionL1"
          args="rotation" />
      </accessPoint>
    </accessModel>
    <semanticModel>
      <symbol name="example_symbol">
        <argument
          class="reve.scene.item.semantic.argument.X3DConnectorL1"
          args="top, translation"/>
      </symbol>
    </semanticModel>
  </item>
</itemGroup>
```

Η γλώσσα VERL αποτελείται από ένα ανεξάρτητο από την εφαρμογή και την υλοποίηση μέρος το οποίο αντιστοιχεί στο επίπεδο σύνταξης XML, και από ένα εξειδικευμένο μέρος το οποίο αντιστοιχεί στις τιμές συγκεκριμένων ορισμένων. Με την συγκεκριμένη δομή η VERL επιτυγχάνει να γεφυρώσει την εννοιολογική, υψηλού επιπέδου περιγραφή ενός εικονικού κόσμου, με την τεχνική, χαμηλού επιπέδου υλοποίηση του, στα πλαίσια ενός συγκεκριμένου εικονικού περιβάλλοντος δια τηρώντας όμως τη διάκριση μεταξύ των δύο πλευρών. Η γλώσσα VERL ξεκινάει ανοίγοντας το tag "world" μέσα στο οποίο θα ορίσουμε όλα τα items με τις ιδιότητες. Στο tag "world" μπορούμε να ορίσουμε χαρακτηριστικά όπως η version του κόσμου που έχουμε δημιουργήσει, το όνομα του κόσμου αλλά και τις διαστάσεις του. Επίσης σε αυτό το σημείο μπορούμε να ορίσουμε και το tag "info" στο οποίο μπορούμε να καταχωρίσουμε κάποια

στοιχεία για το πρόγραμμα μας όπως είναι ο συντάκτης, τα στοιχεία του ημερομηνία και η έκδοση του εικονικού κόσμου [52].

Παράδειγμα ορισμού ενός tag «world» και «info»:

```
<abr version="1.0">
<world version="5.0" name="castle_example" dimensions="">
<info
author="George S"
date="28/10/2013"
revision="1"
email="@hotmail.gr"
organization="University of Piraeus"
description="Castle_example"
/>
.....
.....
.....
</world>
</abr>
```

Η γλώσσα VERL λειτουργεί ως ένα προτυποποιημένο και διαφανές σημείο επαφής μεταξύ του εννοιολογικού επιπέδου και του επιπέδου υλοποίησης, επιτυγχάνοντας ξεκάθαρη διάκριση μεταξύ αυτών των δυο χαρακτηριστικών του εικονικού κόσμου. Στην ουσία, επιτυγχάνει να γεφυρώσει την εννοιολογική, υψηλού επιπέδου περιγραφή ενός εικονικού κόσμου, με την τεχνική, χαμηλού επιπέδου υλοποίησή του, στα πλαίσια ενός συγκεκριμένου εικονικού περιβάλλοντος. Η ανεξαρτησία του εννοιολογικού επιπέδου αντιστοιχεί στο επίπεδο σύνταξης XML, ενώ η υλοποίηση που αφορά την εφαρμογή, αντιστοιχεί στις τιμές συγκεκριμένων ορισμάτων [52].

Η δομή της VERL ορίζεται από ένα DTD (Document Type Definition). Κάθε .verl αρχείο αποτελείται από items. Ως items ορίζονται τα «ξεχωριστά συστατικά ενός εικονικού κόσμου που διακρίνονται από όσο το δυνατόν απλούστερη γεωμετρική δομή, μέγιστη σημασιολογική και αλληλεπιδραστική αυτονομία, καθώς και γενικότερα, ως προς το ρόλο τους στον εικονικό κόσμο» [52]. Αν τα αντικείμενα αυτά βρίσκονται στην ίδια τοποθεσία προέλευσης, δηλαδή στο ίδιο VRML/X3D αρχείο, τότε υπάγονται και στο ίδιο itemGroup, ενώ σε διαφορετική περίπτωση υπάγονται σε διαφορετικά itemGroup [52].

Κάθε itemGroup έχει 3 attributes:

- **name:** το όνομα του itemGroup, όπως το ορίζει ο προγραμματιστής
- **location:** η τοποθεσία του αρχείου VRML ή X3D στο οποίο υπάγονται τα αντικείμενα που μας ενδιαφέρουν
- **behaviours:** Παίρνει τιμές true/false ανάλογα αν θα οριστεί συμπεριφορά για αυτό το itemGroup.

Κάθε item της VERL αντιστοιχεί σε έναν κόμβο κάποιου VRML ή X3D αρχείου, το οποίο ορίζεται και διευκρινίζεται στο itemGroup. Το item έχει επίσης 4 attributes:

- **name:** το όνομα του item, όπως το ορίζει ο προγραμματιστής
- **class:** Παίρνει τιμές real ή unreal ανάλογα αν ο πράκτορας θα αλληλεπιδράσει με αυτό το αντικείμενο ή όχι

- **fit**: Παίρνει τιμές true ή false, ανάλογα αν θέλουμε το τοπικό σύστημα συντεταγμένων του αντικειμένου να ευθυγραμμίζεται με το σύστημα συντεταγμένων του κόσμου ή όχι
- **fitcentre**: Παίρνει τιμές true ή false, ανάλογα αν θέλουμε το κέντρο του boundingbox του αντικειμένου να συμπίπτει με την αρχή των αξόνων του συστήματος συντεταγμένων του κόσμου.

Η δομή, επομένως, ενός itemGroup έχει την παρακάτω μορφή:

```
<itemGroup name="<itemGroupName>" location="<location>" behaviours="true">
<item name="<itemName>" class="<itemClass>" fit="true" fitCentre="false">
</item>
</itemGroup>
```

Κάθε ένα από αυτά τα itemGroups ή τα items μπορούν να αποδοθούν συγκεκριμένα χαρακτηριστικά όπως π.χ. class: real/unreal, behaviors: true/false.

Επιπρόσθετα αναπαριστούνται τα τρία item aspects (Physical aspect, Semantic aspect, Virtual aspect).

Για κάθε item, η VERL προσφέρει 3 επίπεδα ορισμού μοντέλων: virtual model, semantic model και access model definition, όπου αντιστοιχούν στις διάφορες όψεις που έχει ένα αντικείμενο. Πιο συγκεκριμένα, το virtual model αφορά στη φυσική αναπαράσταση του αντικειμένου (physical aspect), το semantic model αφορά στην σημασιολογική άποψη του αντικειμένου (semantic aspect), που επιτυγχάνεται με την χρήση συμβόλων, ενώ το access model αφορά στις δυνατότητες δράσης του αντικειμένου και αλληλεπίδρασης με αυτό, με τη χρήση σημείων πρόσβασης (accesspoints) και μεθόδων(functions) που προσδίδουν τη λειτουργικότητα [52].

Στο virtual model definition υπάρχει το attribute *source* στο οποίο ορίζεται η πηγή του αντικειμένου, δηλαδή το όνομα που έχει ο κόμβος που μας ενδιαφέρει στο αντίστοιχο VRML/X3D αρχείο. Προϋπόθεση, λοιπόν, συμμετοχής ενός κόμβου στην αναπαράσταση REVE είναι να πρόκειται για έναν ονοματισμένο DEF κόμβο στο αντίστοιχο VRML/X3D αρχείο. Επίσης, στο virtual model περιλαμβάνεται ο ορισμός του στοιχείου *<transform>*, όπου ορίζονται πιθανοί μετασχηματισμοί του αντικειμένου στο χώρο και πιο συγκεκριμένα περιλαμβάνει τα attributes:

- **translation**: μετατόπιση του αντικειμένου στους άξονες xyz
- **rotation**: περιστροφή του αντικειμένου βάσει κάποιου άξονα κατά ακτίνια και
- **scale**: μεταβολή του μεγέθους ενός αντικειμένου ανά άξονα.

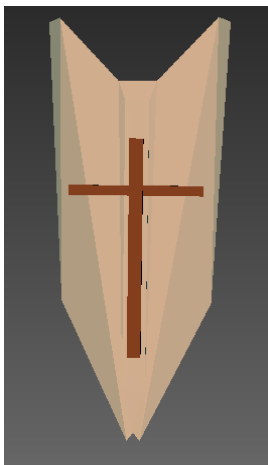
Η δομή, λοιπόν, του virtual model ενός item έχει την παρακάτω μορφή:

```
<item name="<itemName>" class="<itemClass>" fit="true" fitCentre="false">
<virtualModel source="<virtualModelSource>">
<transform translation="<tx ty tz>" rotation="<rx ry rz ra>" scale="<sx sy sz>" />
</virtualModel>
</item>
```

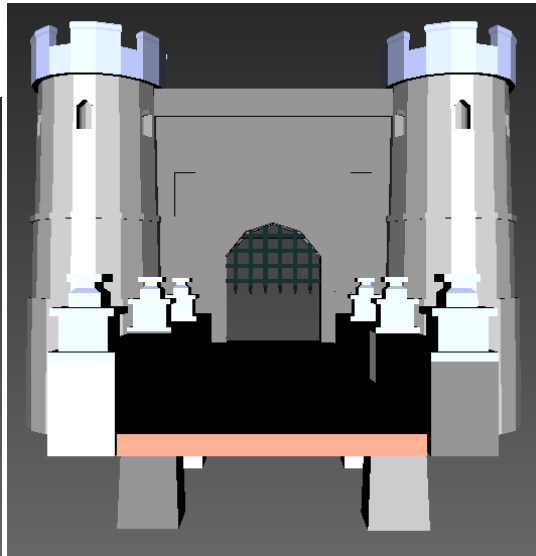
Ένα παράδειγμα χρήσης του virtualmodel είναι το παρακάτω, όπως το χρησιμοποιήσαμε στην μεταπτυχιακή μας εργασία στο μάθημα 'Ευφυή Εικονικά Περιβάλλοντα' για την τοποθέτηση δυο ασπίδων στον εικονικό κόσμο, έτσι ώστε να συμπέσουν πάνω στις πύλες ενός κάστρου:

```
<itemGroup name="shield" location="knight_shield.wrl" behaviours="true">
<item name="shield1" class="real.general" fit="true">
<virtualModel source="Shield001">
```

```
<transform translation="4.5 9 3.5" rotation="0.4 0 0 1.5" scale="0.2 0.2 0.2"/>
</virtualModel>
</item>
<item name="shield2" class="real.general" fit="true">
<virtualModel source="Shield001">
<transform translation="18.5 9 3.5" rotation="0.4 0 0 1.5" scale="0.2 0.2 0.2"/>
</virtualModel>
</item>
</itemGroup>
```

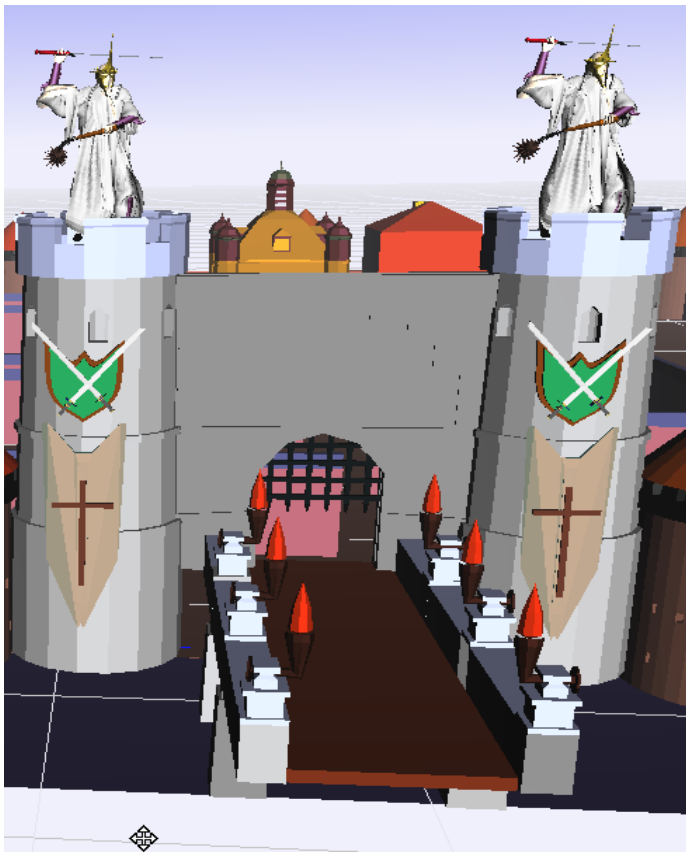


Εικόνα 55: (α) ασπίδα (wrl)



β) πύλη (wrl)





Εικόνα 56: Δημιουργία πύλη με ασπίδες μέσω veril

Όπως αναφέρθηκε, η VERL δίνει τη δυνατότητα ορισμού και ενός σημασιολογικού μοντέλου (semantic model) για κάθε αντικείμενο. Πιο συγκεκριμένα, για αυτόν τον σκοπό μέσα σε κάθε item μπορούμε να χρησιμοποιήσουμε ένα στοιχείο <semanticModel>, το οποίο παρέχει πληροφορίες για κάθε αντικείμενο με τη μορφή συμβόλων (symbols). Τα σύμβολα διακρίνονται σε δυο κατηγορίες: α) αυτά που είναι προκαθορισμένα και αφορούν γενικές πληροφορίες για το αντικείμενο, όπως η κλάση του, η τοποθεσία του, ο προσανατολισμός του και οι διαστάσεις του boundingbox του και β) σε αυτά που μπορούν να οριστούν στην εφαρμογή, δίνοντας επιπλέον δυνατότητες σε κάθε αντικείμενο. Στην ουσία, ο ορισμός συμβόλων για κάθε είδους σημασιολογία για ένα item, επιτρέπει στο χρήστη να αντιμετωπίζει όλα τα πιθανά προβλήματα ενός μοντέλου.

Κάθε στοιχείο <symbol> έχει ένα attribute name, όπου ορίζεται η ονομασία του. Κάθε item μπορεί να έχει ένα ή παραπάνω symbol, που εμπεριέχει στοιχεία τύπου <argument>, τα οποία με τη σειρά τους λαμβάνουν τα παρακάτω 2 attributes:

- **class:** Σε επίπεδο προδιαγραφών, πρόκειται για τον σημασιολογικό ορισμό (semantics) των παραμέτρων (args) που ακολουθούν. Σε επίπεδο υλοποίησης, προσδιορίζει την java κλάση ενός αντικειμένου που θα παρέχει τιμή για την παράμετρο.
- **args:** Πρόκειται για τις παραμέτρους του συγκεκριμένου <argument>, που αν είναι παραπάνω από μια, διακρίνονται με κενό ή κόμματα. Η τιμή αυτού του attribute παίρνει ως string τις παραμέτρους που παράγει η μέθοδος δημιουργίας του αντικειμένου που προσδιορίζεται στην class και παρέχει τιμή για την παράμετρο.

Η αναπαράσταση REVE προσφέρει 2built-inargumentclasses, οι οποίες είναι οι:

- **X3DConnectorL1:** Για αυτή την κλάση, η τιμή της παραμέτρου αντιστοιχεί στην τιμή ενός συγκεκριμένου πεδίου ενός κόμβου στην ιεραρχία του virtual model. Το πεδίο πρέπει να είναι προσβάσιμο, είτε ως τύπος εισόδου είτε ως τύπος εξόδου και η τιμή του προωθείται ως έχει, χωρίς μετατροπές, παρά μόνο περικλείοντάς την σε string. Σε περίπτωση που η τιμή του πεδίου δεν είναι ορισμένη ή προκύψει κάποιο σφάλμα, η παράμετρος παίρνει την τιμή “<null>”. Τέλος, κάθε φορά που απαιτείται η τιμή της παραμέτρου, η τιμή του πεδίου διαβάζεται εκ νέου, ορίζοντας μια ενεργή σχέση μεταξύ του πεδίου και της παραμέτρου. Η πλήρης ονομασία της κλάσης είναι *reve.scene.item.semantic.argument.X3DConnectorL1*
- **ZeroArgumentItemMethodConnectorL1:** Η τιμή της παραμέτρου για αυτή την κλάση είναι η τιμή που επιστρέφει μια συγκεκριμένη public μέθοδος του item του οποίου το semantic model εμπεριέχει το symbol. Αντίστοιχα και εδώ, η τιμή δεν επιδέχεται μετατροπές και περνιέται ως string. Αν η τιμή επιστροφής είναι null ή σε περίπτωση σφάλματος η τιμή του αντικειμένου είναι “<null>” και ανανεώνεται κάθε φορά που ζητείται, τρέχοντας εκ νέου τη μέθοδο για την τιμή επιστροφής. Η πλήρης ονομασία της κλάσης είναι *reve.scene.item.semantic.argument.ZeroArgumentItemMethodConnectorL1*

Η δομή, λοιπόν, του σημασιολογικού μοντέλου έχει την παρακάτω μορφή:

```
<item name="<itemName>" class="<itemClass>" fit="true" fitCentre="false">
<semanticModel>
<symbol name="<symbolName>">
<argument class="<argumentClass>" args="<argumentInitializationArgs>" />
</symbol>
</semanticModel>
</item>
```

Τέλος, η VERL παρέχει και ένα μοντέλο πρόσβασης (access model) όπως αναφέρθηκε. Η εισαγωγή αυτού του μοντέλου επιτρέπει την πρόσθεση λειτουργικότητας σε ένα αντικείμενο και συνεπώς, τη διάδραση ενός χρήστη ή ενός εικονικού πράκτορα με αυτό. Η συμπερίληψη του access model γίνεται με την πρόσθεση του στοιχείου *<accessModel>* εντός ενός στοιχείου *<item>*.

Στη συνέχεια, κάτω από το στοιχείο *<accessModel>* εισάγεται ένα σημείο πρόσβασης (accesspoint), μέσα από το οποίο επιτυγχάνεται η αλληλεπίδραση με το αντικείμενο. Στην ουσία, το σημείο πρόσβασης λειτουργεί ως συνδετικός κρίκος μεταξύ του αντικειμένου και της λειτουργικότητάς του. Για παράδειγμα, ένα accesspoint μπορεί να είναι ένα κουμπί ενεργοποίησης ή απενεργοποίησης μιας κίνησης. Κάθε σημείο πρόσβασης αναπαρίσταται και ως αντικείμενο στον κόσμο, επομένως διακρίνεται από τα χαρακτηριστικά της θέσης, του προσανατολισμού και των διαστάσεων, με την ιδιαιτερότητα πως αυτά ορίζονται πάντα σε σχετική θέση με το τοπικό σύστημα συντεταγμένων του αντικειμένου για το οποίο προορίζεται. Κατά αυτόν τον τρόπο, η μεταβολή των χαρακτηριστικών του αντικειμένου θα παρασύρει και το αντίστοιχο σημείο πρόσβασης, οπότε π.χ. αν ένα κουμπί βρίσκεται πάνω στο αντικείμενο και αυτό μετακινηθεί, αντίστοιχα θα μετακινηθεί και το κουμπί, διατηρώντας τη λογική τους σχέση στο χώρο.

Ο ορισμός ενός σημείου πρόσβασης γίνεται με το στοιχείο *<accessPoint>* που παίρνει τα παρακάτω 2 attributes:

- **name:** το όνομα που προσδίδουμε στο σημείο πρόσβασης
- **node:** το όνομα του κόμβου στο virtual model του αντικειμένου, με το οποίο θα συσχετιστεί το εν λόγω σημείο πρόσβασης

Τέλος, για να προστεθεί και η λειτουργικότητα που συνεπάγεται το σημείο πρόσβασης, δηλαδή τι ενέργεια θα εκτελεστεί στο αντικείμενο εφόσον ο πράκτορας ή ο χρήστης αλληλεπιδράσει με αυτό, π.χ. με το πάτημα ενός κουμπιού το αντικείμενο να μετακινηθεί 10 μέτρα στον άξονα x, ορίζουμε το στοιχείο <function> που παίρνει τα παρακάτω 3 attributes:

- **name:** το όνομα της μεθόδου που εκτελείται
- **class:** η κλάση στην οποία ανήκει η μέθοδος που εκτελείται. Σε επίπεδο προδιαγραφών, πρόκειται για τον ορισμό του τρόπου με τον οποίο θα τροποποιηθεί ένα αντικείμενο, μόλις εκτελεστεί η μέθοδος. Σε επίπεδο υλοποίησης, προσδιορίζει την java κλάση που αποτελεί την υλοποίηση της μεθόδου.
- **args:** αφορά τις παραμέτρους αρχικοποίησης που μπορεί να υπάρχουν για τη συγκεκριμένη μέθοδο

Η αναπαράσταση REVE προσφέρει 3 built-infunction κλάσεις για τη χρήση τους στη VERL, οι οποίες είναι οι εξής:

- **AccessPointTranslateFunctionL1:** η κλάση αυτή χρησιμοποιείται για να μετατοπίσει ένα αντικείμενο σε σχέση με το σημείο πρόσβασής του. Η μετατόπιση μπορεί να είναι είτε απόλυτη, είτε σχετική και ο κόμβος που σχετίζεται με το accesspoint πρέπει να υποστηρίζει μετατόπιση, συνεπώς, να είναι Transform κόμβος. Δεν υπάρχουν παράμετροι αρχικοποίησης για αυτή την κλάση, παρά μόνο παράμετροι εκτέλεσης της μορφής: '<x y z>, <relative>' που δηλώνουν αν η μετατόπιση θα είναι απόλυτη ή σχετική (τιμή false ή true αντίστοιχα) και κατά πόσο (σε μέτρα) για κάθε άξονα. Το πλήρες όνομα της κλάσης είναι `reve.scene.item.access.function.AccessPointTranslateFunctionL1`.
- **AccessPointRotateFunctionL1:** η κλάση αυτή χρησιμοποιείται για να περιστρέψει ένα αντικείμενο σε σχέση με το σημείο πρόσβασής του. Η περιστροφή μπορεί να είναι είτε απόλυτη, είτε σχετική και ο κόμβος που σχετίζεται με το accesspoint πρέπει να την υποστηρίζει, συνεπώς, να είναι Transform κόμβος. Δεν υπάρχουν παράμετροι αρχικοποίησης για αυτή την κλάση, παρά μόνο παράμετροι εκτέλεσης της μορφής: '<x y z a>, <relative>' που δηλώνουν αν η περιστροφή θα είναι απόλυτη ή σχετική (τιμή false ή true αντίστοιχα), σε ποιον άξονα θα γίνει (τιμή 1 ή 0) και κατά πόσο (σε ακτίνια) για κάθε άξονα. Το πλήρες όνομα της κλάσης είναι `reve.scene.item.access.function.AccessPointRotateFunctionL1`
- **X3DFieldFunctionL1:** η κλάση χρησιμοποιείται για να γράψει μια τιμή σε ένα συγκεκριμένο πεδίο ενός συγκεκριμένου κόμβου στο virtualmodel του αντικειμένου. Προϋπόθεση αποτελεί το πεδίο αυτό να είναι εγγράψιμο, δηλαδή ο τύπος πρόσβασής του να είναι είτε "inputOutput" είτε "inputOnly". Η μέθοδος προσπαθεί να μετατρέψει την δοσμένη τιμή στον τύπο του πεδίου. Αν η μετατροπή αποτύχει, η εκτέλεση σταματάει. Οι παράμετροι αρχικοποίησης είναι της μορφής: '<node\_name>, <field\_name>', ενώ οι παράμετροι εκτέλεσης εξαρτώνται από τον τύπο του πεδίου. Το πλήρες όνομα της κλάσης είναι `reve.scene.item.access.function.X3DFieldFunctionL1`.

Η δομή, λοιπόν, του μοντέλου αλληλεπίδρασης έχει την παρακάτω μορφή:

```
<item name="<itemName>" class="<itemClass>" fit="true" fitCentre="false">
<accessModel>
<accessPoint name="<accessPointName>" node="<nodeName>">
<function name="<functionName>" class="<functionClass>" args="<functionInitializationArgs>"
/>
</accessPoint>
```

```
</accessModel>
</item>
```

Ένα παράδειγμα χρήσης του access model προκύπτει από την μεταπτυχιακή μας εργασία για το μάθημα 'Ευφυή Εικονικά Περιβάλλοντα', όπου διάφοροι πράκτορες περιπλανώνται σε έναν εικονικό κόσμο, εις αναζήτηση μιας βόμβας. Η βόμβα μας είχε έναν μοχλό που κινούταν όσο η βόμβα ήταν ενεργή. Όταν ο πράκτορας-πυροτεχνουργός έβρισκε τη βόμβα, τοποθετούσε το χέρι του πάνω σε ένα κουμπί (το access point) για να την απενεργοποιήσει, γεγονός που απεικονιζόταν με το σταμάτημα του μοχλού της βόμβας. Για να επιτύχουμε αυτό το αποτέλεσμα, στο .x3d αρχείο μας με τον κόμβο Bomba, τοποθετήσαμε μέσα στα ομαδοποιημένα προς κίνηση αντικείμενα (τον μοχλό) μία συνάρτηση «SpeakerVibration-TS TimeSensor» και στο τέλος του αντικείμενου, ένα διακόπτη, τον «MultiSwitch-SC Script», για την διαδραστικότητα της απενεργοποίησης της βόμβας, μέσω μίας συνάρτησης (της setActive). Για τον ορισμό της λειτουργικότητας στο συγκεκριμένο αντικείμενο, ορίσαμε το παρακάτω .xml αρχείο:

```
<abr version="1.0">
  <world version="1.1" name="key_bombTest" dimensions="100.0, 261.8">
    <itemGroup name="bomb_example" location="key_bomb.x3dv" behaviours="true">
      <item name="bomb" class="real.general" fit="true">
        <virtualModel source="Bomba">
          <transform />
        </virtualModel>
        <accessModel>
          <accessPoint name="buttons" node="MultiSwitch-SC">
            <function
              name="setActive"
              class="reve.scene.item.access.function.X3DFieldFunctionL1"
              args="setActive"/>
          </accessPoint>
        </accessModel>
      </item>
    </itemGroup>
  </world>
</abr>
```

Έτσι, όταν ο πράκτορας έφτανε στη βόμβα, τοποθετούσε το χέρι του στο σημείο πρόσβασης(κουμπί), μέσω του οποίου καλούταν η μέθοδος setActive και σταματούσε την κίνηση του μοχλού.



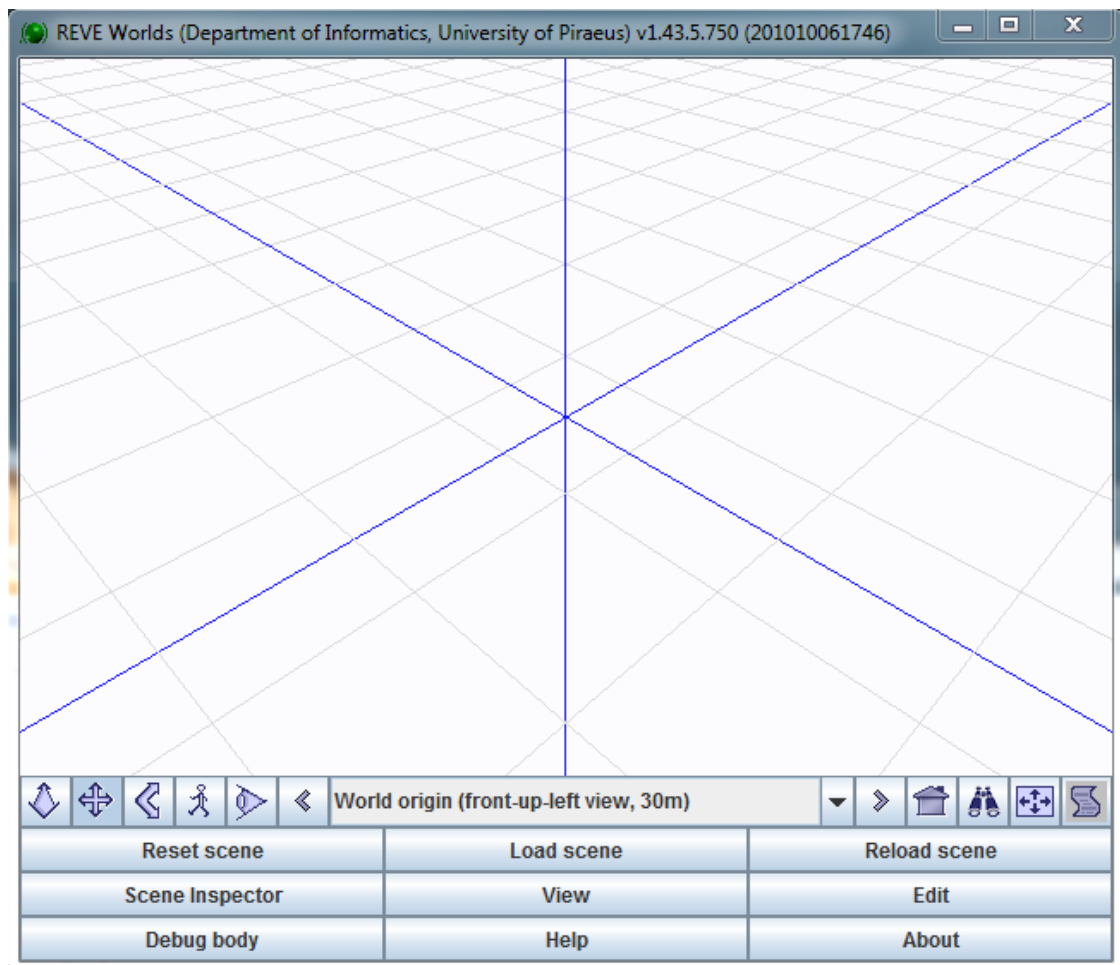
**Εικόνα 57: Βόμβα με μοχλό και κουμπί απενεργοποίησης**

Συμπερασματικά, η VERL, δεδομένου ότι βασίζεται στην μεταγλώσσα XML, διακρίνεται και από τα αντίστοιχα πλεονεκτήματα που αυτή προσφέρει. Συνεπώς, παρέχει τη δυνατότητα υλοποίησης σε διαφορετικές πλατφόρμες, είναι ανεξάρτητη από το περιβάλλον εκτέλεσης, είναι απλή και φιλική προς το χρήστη, είναι επεκτάσιμη και αποτελεί ένα ισχυρό, εκφραστικά, εργαλείο για τη μοντελοποίηση εικονικών κόσμων σύμφωνα με την αναπαράσταση REVE. Η VERL, ως εκ τούτου, μπορεί να γίνει αντικείμενο επεξεργασίας από πλήθος διαθέσιμων σχεδιαστικών εργαλείων και βιβλιοθηκών λεκτικής και συντακτικής επεξεργασίας, διατηρώντας ξεκάθαρη και ανεπηρέαστη, σε κάθε περίπτωση, τη σημασιολογία την οποία αναπαριστά, καθιστώντας την ιδανική για αναπαραστάσεις εικονικών κόσμων.

## Κεφάλαιο 10 - Η Πλατφόρμα REVEWORLDS

Η πλατφόρμα REVE Worlds είναι μια εφαρμογή που ενσαρκώνει την αναπαράσταση εικονικών κόσμων με βάση την αναπαράσταση REVE, η οποία επιτρέπει μια υψηλού επιπέδου αναπαράσταση του κόσμου, διαχωρίζοντας το εννοιολογικό πλαίσιο από την υλοποίησή του. Πιο συγκεκριμένα, προσεγγίζει γενικές, αλλά ταυτόχρονα, βασικές έννοιες όπως: αντικείμενο, σημασιολογία αντικείμενου, λειτουργικότητα, αλληλεπίδραση, κόσμος, διακρίνοντας τα συστατικά που αποτελούν έναν εικονικό κόσμο από την υλοποίησή του για μια συγκεκριμένη εφαρμογή, γεγονός που κάνει την αναπαράσταση REVE ιδιαίτερα εύχρηστη, ευέλικτη και επεκτάσιμη.

Η εφαρμογή REVE Worlds αποτελείται από μια διεπαφή (API) και από ένα σύνολο συστατικών λογισμικού και βοηθημάτων. Είναι γραμμένη σε γλώσσα java και χρησιμοποιεί τη βιβλιοθήκη Xj3D, η οποία αποτελεί μια υλοποίηση του προτύπου X3D σε java. Ως εκ τούτου, υποστηρίζει αντικείμενα και μοντέλα σκηνής που βασίζονται στο πρότυπο X3D καθώς και στην VRML97, της οποίας αποτελεί εξέλιξη, προσφέροντας έτσι ένα μεγάλο εύρος τρισδιάστατων μοντέλων. Από την άλλη, η γλώσσα VERL αποτελεί το συνδετικό εργαλείο ανάμεσα στην πλατφόρμα REVE Worlds και την αναπαράσταση REVE. Συνεπώς, η αναπαράσταση ενός κόσμου στην πλατφόρμα επιτυγχάνεται με το άνοιγμα ενός αρχείου .verl σε αυτήν, κατά το οποίο φορτώνεται ο κόσμος και εμφανίζεται σε ένα τρισδιάστατο άξονα (πλέγμα) συντεταγμένων.

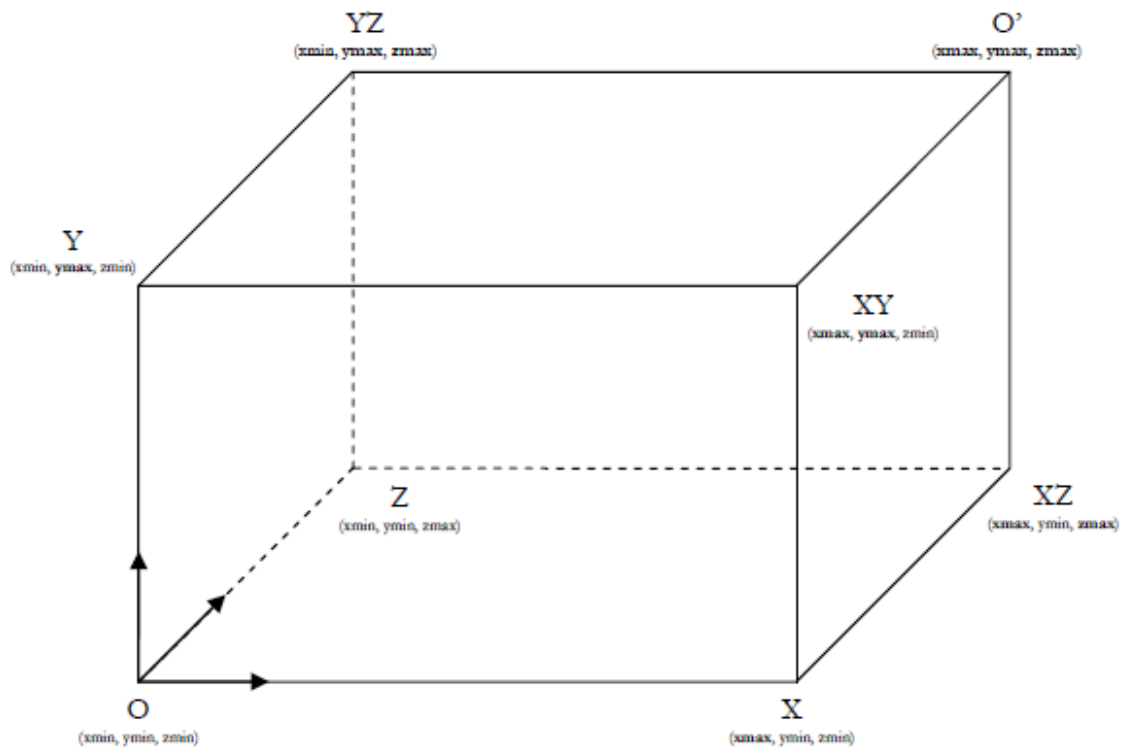


Εικόνα 58: Η πλατφόρμα REVEWorlds

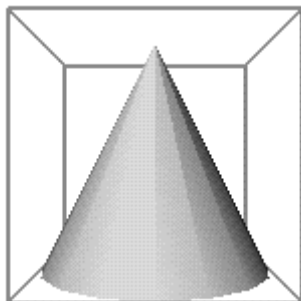
Η πλατφόρμα είναι ιδιαίτερα φιλική προς τον χρήστη, προσφέροντάς του μια ποικιλία δυνατοτήτων μέσω εργαλείων του GUI. Για παράδειγμα, ο χρήστης μπορεί να περιηγηθεί στον εικονικό κόσμο και να έχει την αίσθηση πως είτε περπατάει σε αυτόν είτε τον μελετάει από ψηλά, με την εντύπωση πως πλανιέται από πάνω του. Επίσης, ο χρήστης, μπορεί να μεταφερθεί σε ένα συγκεκριμένο μέρος του χώρου όπου θα επιλέξει, μελετώντας από κοντινή απόσταση αυτό το συγκεκριμένο τμήμα του κόσμου και επιτρέποντας τη γρήγορη μετάβαση από ένα σημείο σε ένα άλλο. Ακόμα, ο χρήστης μπορεί να επιλέξει διαφορετικές οπτικές γωνίες παρατήρησης του κόσμου και να επιστρέψει οποιαδήποτε στιγμή στην πρωταρχική οπτική γωνία, δηλαδή την άποψη που είχε ο κόσμος όταν φορτώθηκε για πρώτη φορά στην πλατφόρμα. Τέλος, ο χρήστης μπορεί να εκτελέσει επιπλέον ενέργειες που του επιτρέπουν να επεξεργαστεί και να βελτιώσει την ανάπτυξη του εικονικού κόσμου, μεταξύ των οποίων είναι η επαναφόρτωση ή ο επαναορισμός μιας σκηνής, η προσθαφαίρεση οπτικών γωνιών παρατήρησης και η μελέτη της δομής του γραφήματος σκηνής και των αντικειμένων/κόμβων, καθώς και των boundingboxes που τα περικλείουν. Επίσης, μέσω εργαλείων της REVE είναι εφικτή και η αλληλεπίδραση με αντικείμενα που χαρακτηρίζονται από κάποια λειτουργικότητα, δηλαδή έχουν accesspoints και functions [52].

Είναι σημαντικό να σημειώσουμε πως στην πλατφόρμα REVEWorlds, κάθε <item> που ορίζεται στην VERL, ενώ εμφανίζεται ως συμπαγές αντικείμενο στον τριδιάστατο χώρο, περικλείεται από ένα προσανατολισμένο ορθογώνιο παραλληλεπίπεδο, μοναδικό για κάθε αντικείμενο και γνωστό στην βιβλιογραφία ως *OrientedBoundingBox* (OBB). Προϋπόθεση σωστού υπολογισμού του boundingbox ενός αντικειμένου είναι η εισαγωγή του στον κόσμο ως συνόλου συνδεδεμένων ευθύγραμμων τμημάτων ή επιφανειών και όχι ως συμπαγών πρωτογενών σχημάτων. Για παράδειγμα, αν έχουμε ένα αντικείμενο-σφαίρα στην VRML, αντί να το φορτώσουμε ως γεωμετρία Sphere, θα πρέπει να το μετατρέψουμε σε γεωμετρία IndexedFaceSet, πράγμα που γίνεται αυτόματα από το εργαλείο επεξεργασίας VRML αντικειμένων, VrmIPad.

Στην ουσία, το *OrientedBoundingBox* είναι ένα τριδιάστατο ορθογώνιο παραλληλεπίπεδο που έχει θέση, διαστάσεις και προσανατολισμό και περιβάλλει κάθε αντικείμενο ξεχωριστά, με το μέγεθός του να είναι το ελάχιστο δυνατό ώστε να περικλείει το εν λόγω αντικείμενο. Πιο συγκεκριμένα, οι διαστάσεις του για κάθε άξονα υπολογίζονται από το ζευγάρι συντεταγμένων με την μέγιστη και ελάχιστη τιμή (ακρότατα) που εμφανίζονται στο item που περικλείεται. Έτσι, για τον άξονα x οι διαστάσεις είναι  $(x_{min}, x_{max})$  του αντικειμένου, για τον άξονα y τα  $(y_{min}, y_{max})$  και για τον άξονα z είναι τα  $(z_{min}, z_{max})$ , έχοντας την μορφή που φαίνεται στην παρακάτω εικόνα:



Εικόνα 59: Διαστάσεις Bounding Box



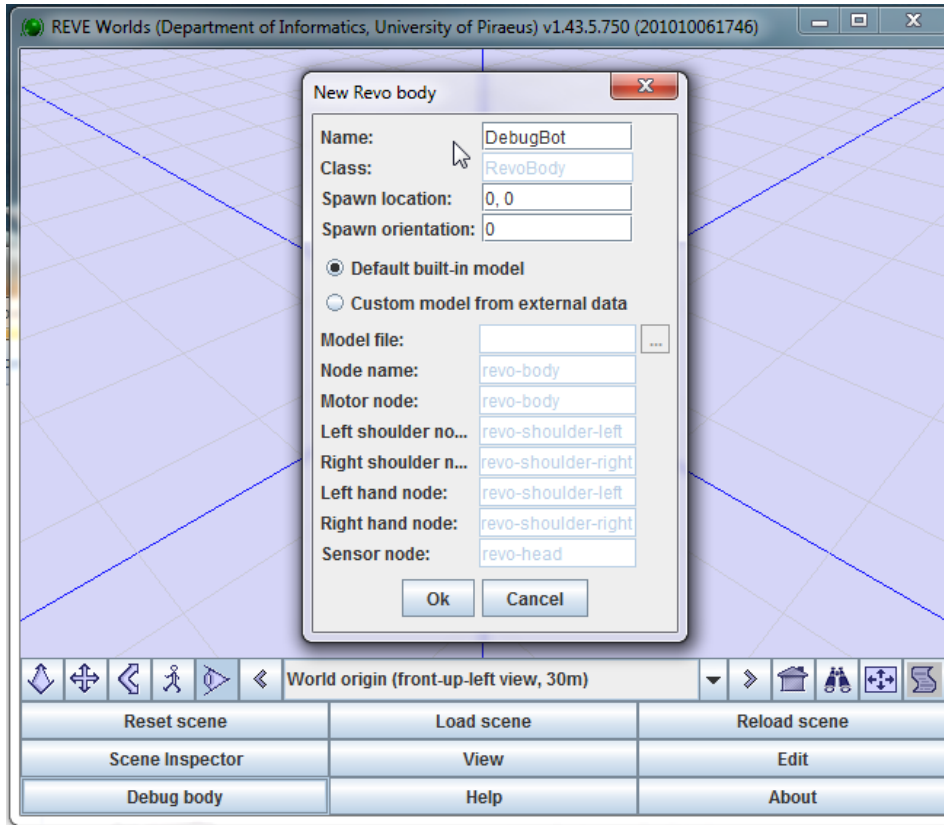
Εικόνα 60: Κώνος με το bounding box του

Τέλος, η πλατφόρμα REVE Worlds προσφέρει τη δυνατότητα εισαγωγής πρακτόρων στη σκηνή, οι οποίοι είναι ικανοί να αλληλεπιδράσουν με τον εικονικό κόσμο, αντιλαμβανόμενοι τα σημεία πρόσβασης (accesspoints) των αντικειμένων και ενεργοποιώντας την όποια λειτουργία τους. Οι εικονικοί πράκτορες είναι σχεδιασμένοι ειδικά για αυτόν τον σκοπό και προσπαθούν να προσομοιώσουν το ανθρώπινο σώμα, έχοντας μάτια για να αντιλαμβάνονται το χώρο που βρίσκεται στο οπτικό τους πεδίο, χέρια για να αλληλεπιδρούν με τα αντικείμενα και σώμα για να περιπλανιούνται στον κόσμο. Η αποφυγή συγκρούσεων των πρακτόρων με τα αντικείμενα του κόσμου επιτυγχάνεται μέσω της αντίληψης των boundingboxes των αντικειμένων, εξού και η γεωμετρία IndexedFaceSet και IndexedLineSet των αντικειμένων είναι σημαντική [52].

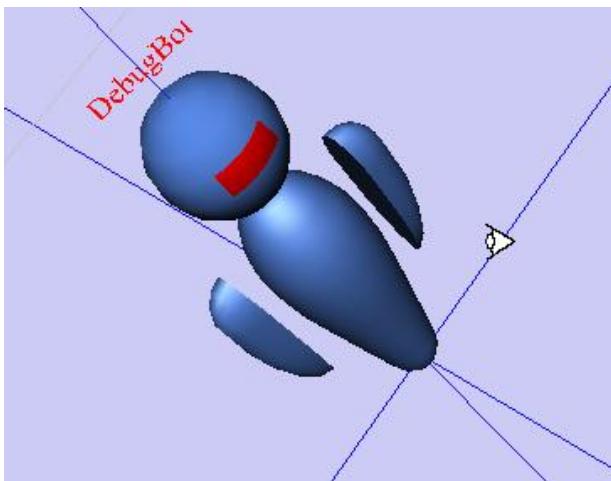
Η αρχικοποίηση και τοποθέτηση ενός πράκτορα στον κόσμο - με τον ορισμό του ονόματός του, της θέσης και του προσανατολισμού του - μπορεί να γίνει είτε χειροκίνητα μέσα από το εργαλείο *Debugbody* της πλατφόρμας, όπου by default ο πράκτορας εμφανίζεται στην αρχή των αξόνων, είτε προγραμματιστικά, με κλάσεις java που υλοποιούν το πακέτο `reve.sara.agent.*` όπου μπορεί να προστεθεί και ευφυΐα στον πράκτορα για μια ροή ενεργειών



που οφείλει να εκτελέσει. Αντίστοιχα, η περιπλάνηση του πράκτορα και η διάδραση με τον κόσμο μπορεί να γίνει είτε χειροκίνητα από τον χρήστη, μέσω μιας γραμμής εντολών CLI για τον έλεγχο της κίνησης του ή προγραμματιστικά με κλάσεις java.



Εικόνα 61: Το εργαλείο Debugbody

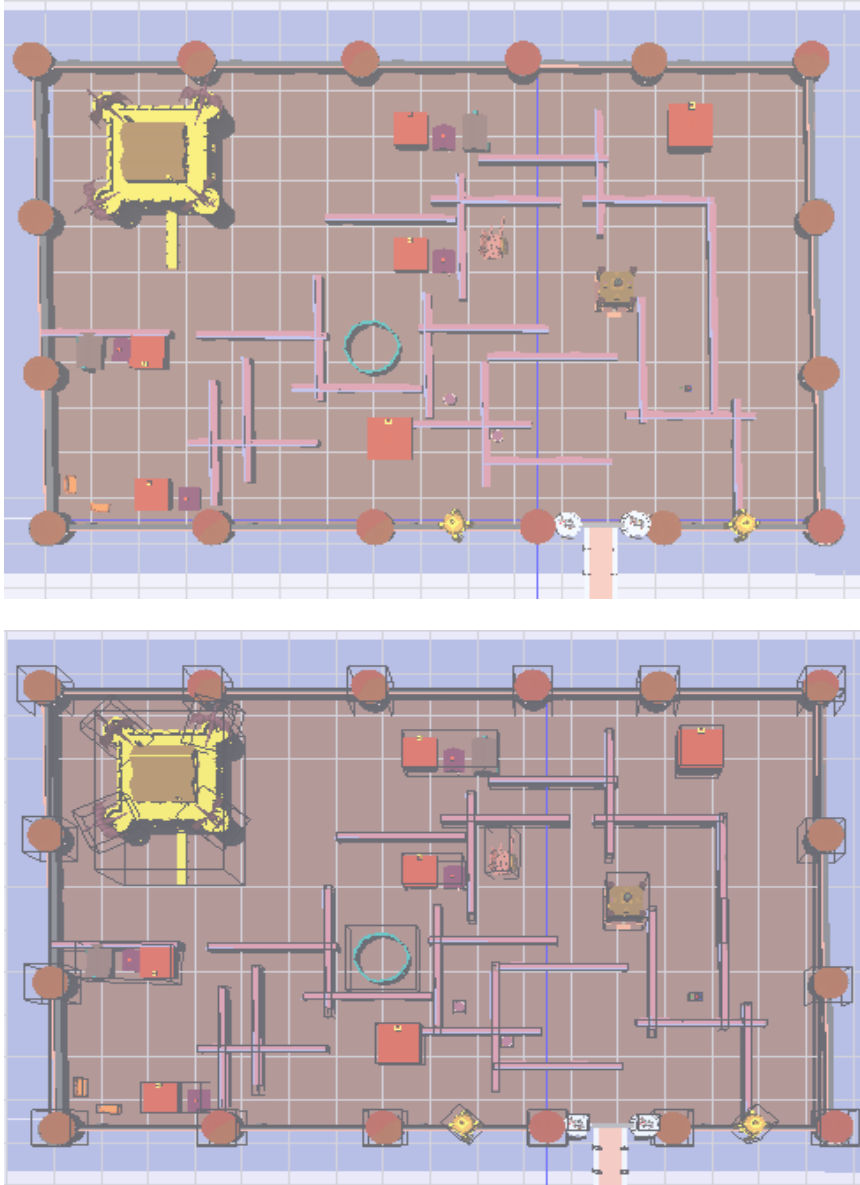


Εικόνα 62: Ο πράκτορας της REVE

Στην ουσία, η REVE περιλαμβάνει εργαλεία για πλήρη έλεγχο ενός πράκτορα, πέρα από τον ορισμό της συμπεριφοράς του, που μπορεί να γίνει προγραμματιστικά σε java. Ωστόσο, το κεφάλαιο των ευφυών εικονικών πρακτόρων της REVE δεν θα μας απασχολήσει περαιτέρω στην εν λόγω διατριβή. Οφείλουμε, όμως, να επισημάνουμε πως οι πράκτορες αντιλαμβάνονται τα bounding boxes των αντικειμένων που είναι ορατά, που δηλώνονται δηλαδή

ως real αντικείμενα στη *verl*, και όχι τα ίδια τα αντικείμενα, γεγονός που θα μας απασχολήσει για τη σωστή σύνθεση του κόσμου στην εφαρμογή, ώστε να αποφεύγονται επικαλύψεις που μπορούν να οδηγήσουν σε σφάλματα. Για την σωστή δημιουργία των *bounding boxes* πρέπει να αποφεύγονται χρήσεις προκαθορισμένων σχημάτων στα αντικείμενα, όπως η σφαίρα, ο κώνος κτλ και να γίνεται μετατροπή της γεωμετρίας τους σε πολύγωνα (*indexedFaceSet* ή *lineSet*) [52].

Τέλος, παρουσιάζουμε ένα παράδειγμα εικονικού κόσμου στην REVE αρχικά χωρίς και στην συνέχεια με τα περικλείοντα *bounding boxes* των αντικειμένων.



Εικόνα 63: Εικονικός κόσμος (α) με BB, (β) χωρίς BB

## Κεφάλαιο 11 - Σχεδίαση Της Εφαρμογής – Ανάλυση Απαιτήσεων

Στην φάση σχεδιασμού του λογισμικού απεικόνισης 3D μοντέλων ήταν απαραίτητη η ανάλυση απαιτήσεων δηλαδή η ανίχνευση των αναγκών τις οποίες θα καλύψει το λογισμικό αλλά και οι δυνατότητες που θα πρέπει να έχει το λογισμικό με σκοπό να θεωρείται πλήρες. Σε αυτό το κεφάλαιο μελετάμε τις απαιτήσεις που θέλουμε να πληρεί η εφαρμογή ReveWorldsDesigner για να είναι εύχρηστη και αξιοποιήσιμη από τους χρήστες. Να υπενθυμίσουμε πως στόχος της εφαρμογής είναι να μπορεί να συνθέσει, να αναπαραστήσει και να επεξεργαστεί έναν εικονικό κόσμο σε γλώσσα νεφί. Επιπλέον σαν στόχος είχε τεθεί η ανάπτυξη ενός λογισμικού χαμηλού σε απαιτήσεις υπολογιστικής ισχύος και συμβατού με ευρέως διαδεδομένων λειτουργικών. Ιδιαίτερη προσοχή δόθηκε στο μέγεθος της εκτελέσιμης εφαρμογής που καταλαμβάνει στην οθόνη του χρήστη ώστε να μπορεί να είναι ευανάγνωστη και σε μικρές οθόνες (laptop-netbook).

Επειδή το λογισμικό απεικόνισης 3D μοντέλων απευθύνεται σε μεγάλο φάσμα ηλικιών ήταν αναγκαία η ανάπτυξη μιας λιτής και φιλικής εφαρμογής προς το χρήστη, χωρίς μεγάλες χρωματικές εναλλαγές, ήχους και πολλά κουμπιά – επιλογές που θα αποσυντονίζουν την διαδικασία επεξεργασίας των 3D μοντέλων. Μια εφαρμογή που δεν θα απωθεί το χρήστη αλλά να τον ενθαρρύνει να ανακαλύπτει όλες τις δυνατότητες του προγράμματος και πάνω από όλα μια εφαρμογή που θα τον βοηθάει να χειρίζεται με ευκολία τα x3d και νεφί αρχεία του.

Όσο φιλικό και να είναι το σύστημα διεπαφής, ο χρήστης θα κάνει λάθη και θα προσπαθήσει να κάνει πράγματα που δε γίνονται μέσα στο πρόγραμμα όπως για παράδειγμα να πατήσει ένα λάθος κουμπί. Γενικά μπορεί να κάνει πράγματα που θα πρέπει να προβλεφθούν στην φάση σχεδιασμού ώστε το σύστημα να είναι ικανό να μην σταματάει την λειτουργία του από πιθανά λάθη, να γνωστοποιεί στο χρήστη τι επιτρέπεται να κάνει και να σχεδιαστεί με τέτοιο τρόπο ώστε να περιορίζεται η πιθανότητα λάθους. Η αποφυγή λαθών επιτυγχάνεται σε μεγαλύτερο βαθμό όταν τα μηνύματα λάθους είναι συγκεκριμένα και περιεκτικά, όταν απενεργοποιούνται εντολές που δεν μπορούν να χρησιμοποιηθούν σε μία συγκεκριμένη περίπτωση και όταν ολοκληρώνονται αυτόματα κάποιες εντολές σε περίπτωση που δεν έχει δοθεί αρκετή πληροφορία από τον χρήστη. Επιπλέον πρέπει να παρέχει στους χρήστες τουλάχιστον τα βασικά εργαλεία τα οποία παρέχουν όλα τα προγράμματα της ίδιας κατηγορίας. Τέτοιες δυνατότητες είναι η εισαγωγή(import) και η εξαγωγή(export) αρχείων σε μορφή .vmtl και .x3d, η δυνατότητα open και save file, η δυνατότητα απεικόνισης της σκηνής γραφήματος σε μορφή δένδρου, η χρήση της drag and drop επιλογής για την μετακίνηση αντικειμένων η προβολή των αντικειμένων από εμπρόσθια κάμερα αλλά και η εναλλαγή της γωνίας απεικόνισης των αντικειμένων.

### 11.1 Ανάλυση Απαιτήσεων

Αρχικά η εφαρμογή πρέπει να είναι ανεξάρτητη από το λειτουργικό σύστημα, δηλαδή να μπορεί να χρησιμοποιηθεί από διαφορετικά λειτουργικά περιβάλλοντα και ταυτόχρονα να μην απαιτεί πολλούς υπολογιστικούς πόρους από άποψη κατανάλωσης μνήμης και επεξεργαστικής ισχύος, έτσι ώστε να μπορεί να γίνει εύκολα προσβάσιμη στους χρήστες από έναν απλό προσωπικό υπολογιστή. Επίσης, θέλουμε ιδανικά η εφαρμογή μας να είναι scalable, δηλαδή να μπορεί να υποστηρίζει πολλαπλούς χρήστες ταυτόχρονα, δίνοντας τη δυνατότητα στο πλήθος των χρηστών να αυξάνεται. Αυτό συνεπάγεται την ταυτόχρονη δημιουργία ενός εικονικού κόσμου από πολλούς χρήστες. Ωστόσο, η συγκεκριμένη λειτουργία αποτελεί δευτερεύων στόχο στην παρούσα διπλωματική διατριβή.

Είναι σημαντικό όλες οι επιθυμητές ενέργειες που στοχεύουμε να πραγματοποιήσει ο χρήστης μέσω της εφαρμογής να γίνονται μέσα από ένα περιβάλλον εύχρηστο, διαδραστικό, κατανοητό στον χρήστη και φιλικό προς αυτόν. Επομένως η εφαρμογή πρέπει να διαθέτει ένα γραφικό περιβάλλον (Graphic User Interface) το οποίο θα χειρίζεται ο χρήστης για τις λειτουργίες που θέλει να εκτελέσει. Συνεπώς, όλες οι ενέργειες που περιγράφονται παρακάτω θα πρέπει να ενσωματώνονται σε αυτό.

Το GUI θα αποτελείται από ένα κεντρικό παράθυρο και θα περιτριγυρίζεται από διάφορα επιμέρους παράθυρα, για να επιτυγχάνεται μεγαλύτερη αυτονομία και λειτουργικότητα.

Ένα παράθυρο πρέπει να λειτουργεί ως μέσο αναζήτησης κι επιλογής του αντικειμένου που θέλουμε να προσθέσουμε στον κόσμο. Θα προβάλλεται μια λίστα δηλαδή με τα προς εισαγωγή αντικείμενα αλλά και με τα αρχεία/φακέλους που έχει ο χρήστης. Ένα δεύτερο μεγαλύτερο παράθυρο πρέπει να λειτουργεί ως η οθόνη όπου αναπαρίσταται και συντίθεται ο εικονικός κόσμος. Αυτό το παράθυρο θα είναι το κεντρικό σημείο της εφαρμογής και θα εμφανίζεται το τελικό αποτέλεσμα που συνθέτει ο χρήστης. Επιπλέον, θα εμφανίζεται ένα τρισδιάστατο πλέγμα αξόνων (x,y,z) στο οποίο θα τοποθετούνται τα αντικείμενα που θέλουμε να αποτελέσουν τον κόσμο. Το πλέγμα συνίσταται να είναι παρόμοιο, όσον αφορά τις αποστάσεις, με αυτό που υπάρχει στο REVEWorlds, ώστε η μετάβαση από τη μια πλατφόρμα στην άλλη να γίνεται ομαλά, χωρίς επιπτώσεις στον κόσμο και ο χρήστης να έχει εξαρχής υπόψη του την τελική μορφή αναπαράστασης του κόσμου.

Δεδομένου ότι θέλουμε η εφαρμογή να υλοποιεί τις ενέργειες που ως τώρα επιτυγχάνονται μέσω συγγραφής κώδικα VERL, πρέπει, να επιτρέπεται η αναπαράσταση ενός εικονικού κόσμου που δομείται σύμφωνα με την αναπαράσταση REVE και να μπορεί να συνθέσει έναν εικονικό κόσμο από τρισδιάστατα αντικείμενα που η δομή τους ιεραρχείται σε γράφημα σκηνής. Ως εκ τούτου, η εφαρμογή πρέπει από τη μια, να μπορεί να εισάγει και να διαβάζει αρχεία τύπου X3D και VRML, ώστε να χρησιμοποιεί τέτοια αντικείμενα για τη σύνθεση του κόσμου, όπως και VERL αρχεία για να τροποποιεί έναν κόσμο και, από την άλλη, να παράγει/εξάγει VERL αρχεία που αναπαριστούν έναν εικονικό κόσμο και μπορούν να χρησιμοποιηθούν στην πλατφόρμα REVE Worlds. Για αυτό τον σκοπό, είναι απαραίτητο η εφαρμογή να διαβάζει και να επεξεργάζεται τόσο γραφήματα σκηνής όσο και XML κώδικα, πάνω στον οποίο έχει βασιστεί η VERL.

Πέρα από το άνοιγμα των αρχείων, πρέπει η εφαρμογή να δίνει στο χρήστη τη δυνατότητα αναζήτησης φακέλων στον υπολογιστή του, που μπορεί να εμπεριέχουν αντικείμενα για τον κόσμο, εμφανίζοντάς του, μέσω μιας δομή τύπου δένδρου, τους διαθέσιμους φακέλους τους οποίους μπορεί να αναπτύξει για να εμφανιστούν τα ονόματα των αντικειμένων που μπορεί να χρησιμοποιήσει. Επίσης, πατώντας το όνομα ενός αντικειμένου ο χρήστης θα πρέπει να μπορεί να βλέπει σε κάποιο παράθυρο μια μικρογραφία της εικόνας του, ώστε να αποφασίσει αν το χρειάζεται στην προκειμένη περίπτωση για τον κόσμο του. Συνεπώς, αν ο χρήστης θέλει να προσθέσει ένα αντικείμενο στον κόσμο, θα πρέπει να μπορεί να το κάνει είτε, όπως αναφέρθηκε, ανοίγοντας το αρχείο ενός αντικειμένου και εισάγοντάς το στον κόσμο σε μια προκαθορισμένη θέση, όπως η αρχή των αξόνων, είτε επιλέγοντάς το αντικείμενο από το συγκεκριμένο παράθυρο και σέρνοντάς το στο παράθυρο του κόσμου στη θέση που τον ενδιαφέρει. Αντίστοιχα, μέσα από το ίδιο παράθυρο μπορεί να δίνεται η δυνατότητα στο χρήστη να δημιουργεί αντίγραφο ενός αντικειμένου από έναν φάκελο σε έναν άλλο ή αποκοπή ενός αντικειμένου από ένα φάκελο σε έναν άλλον, προσφέροντας καλύτερη διαχείριση των projects του χρήστη.

Μια σημαντική λειτουργία, μετά την εισαγωγή ενός αντικειμένου στον κόσμο, είναι η επεξεργασία του. Με αυτό εννοούμε τις δυνατότητες που προσφέρει το `virtualmodelaspect` στη VERL, δηλαδή η περιστροφή, η μετατόπιση ή η αλλαγή της κλίμακας ενός αντικειμένου που πρέπει να επιτυγχάνει και η εφαρμογή μας. Στόχος είναι αυτές οι ενέργειες να δύνανται με δυο τρόπους. Ο πρώτος θα επιτρέψει την εκτέλεση αυτών των ενεργειών μέσα από το παράθυρο απεικόνισης του κόσμου, με τη βοήθεια του ποντικιού (πχ σύρσιμο ενός αντικειμένου παράλληλα με κάποιον άξονα για μετατόπιση) ή πλήκτρων από το πληκτρολόγιο, εφόσον έχει επιλεγεί το αντικείμενο που μας ενδιαφέρει. Η δεύτερη αφορά την ύπαρξη ενός ξεχωριστού παραθύρου που θα αφορά συγκεκριμένα το `modelaspect` του αντικειμένου, όπου θα εμφανίζονται οι παράμετροι κάθε `<transform>` ενέργειας, συνεπώς ο χρήστης αλλάζοντας τις τιμές αυτών των παραμέτρων θα μπορεί να μετατοπίσει, περιστρέψει ή μεγεθύνει/μικρύνει το επιλεγμένο αντικείμενο αντίστοιχα.

Σε αυτό το σημείο οφείλουμε να τονίσουμε πως η επιλογή ενός αντικειμένου πρέπει να γίνεται εμφανής στο χρήστη. Αυτό σημαίνει πως κατά την επιλογή ενός αντικειμένου, πχ πατώντας ένα αντικείμενο με το ποντίκι, αυτό θα αποκτά διακριτά χαρακτηριστικά, όπως να γίνεται πιο έντονο το περίγραμμά του, ώστε ο χρήστης να είναι σίγουρος πως έχει επιλέξει το σωστό αντικείμενο. Επίσης, μπορεί να υπάρχει μια μικρή περιοχή στο GUI όπου θα εμφανίζεται

σε μικρογραφία το επιλεγμένο αντικείμενο και θα περιστρέφεται, ώστε ο χρήστης να έχει μια περίοπτη άποψη για αυτό.

Επιπλέον, μπορεί να δίνεται η δυνατότητα στον χρήστη να χρησιμοποιήσει πολλές φορές το ίδιο αντικείμενο, δημιουργώντας αντίγραφα του στον κόσμο. Για παράδειγμα, επιλέγοντας ένα αντικείμενο και πατώντας δεξί κλικ με το ποντίκι μπορεί να εμφανίζεται μια επιλογή 'Copy' που να φτιάχνει ένα πανομοιότυπο <item> στον κόσμο, με το ίδιο όνομα και κάποιον αριθμό που θα αντιστοιχεί στον αριθμό αντιγράφου. Αυτό θα διευκολύνει τον χρήστη από τη χειροκίνητη δημιουργία ίδιων αρχείων για να χρησιμοποιήσει πολλαπλές φορές ένα αντικείμενο στον κόσμο. Από την άλλη, πέρα από την εισαγωγή και την τροποποίηση ενός αντικειμένου, θα υπάρχει και η δυνατότητα μετονομασίας ενός αντικειμένου και διαγραφής του, με την έννοια της αφαίρεσής του ως <item> από τον συγκεκριμένο εικονικό κόσμο. Αυτό, αντίστοιχα μπορεί να γίνεται με ενέργεια «Delete» που εμφανίζεται κατά την επιλογή ενός αντικειμένου στην σκηνή.

Επιπλέον, στον χρήστη πρέπει να δίνεται η δυνατότητα να αναιρέσει μια ενέργεια ή να επανεκτελέσει μια ενέργεια (undo-redo) σε περίπτωση που έχει εκτελέσει μια λανθασμένη ενέργεια ή που το προσωρινό αποτέλεσμα δεν του αρέσει, συνεπώς πρέπει, τεχνικά, να αποθηκεύονται σε μια στοιβα οι τελευταίες ενέργειες που πραγματοποιεί, ώστε να ανακαλούνται. Ακόμα, θέλουμε οι πιο συχνές λειτουργίες που θα κάνει ο χρήστης (πχ. άνοιγμα αρχείου, αποθήκευση κόσμου ή αποθήκευση ως, ανίρεση, επανεκτέλεση, αντιγραφή αντικειμένου, επικόλληση αντικειμένου, διαγραφή αντικειμένου) να είναι συγκεντρωμένες και εύκολα προσβάσιμες από τον χρήστη σε ένα συγκεκριμένο χώρο της εφαρμογής, επομένως, να εμπεριέχονται κατά προτίμηση σε ένα toolbar.

Επιστρέφοντας στις δυνατότητες της VERL, θέλουμε μέσα από την εφαρμογή να επιτρέπεται ο χειρισμός κάθε άποψης (aspect) του αντικειμένου, διατηρώντας την αυτονομία μεταξύ των επιπέδων αυτών. Συνεπώς, πέρα από ένα παράθυρο για το virtualaspect, όπως περιγράφηκε παραπάνω, θέλουμε να έχουμε άλλα δυο παράθυρα για το semanticaspect και το accessmodelaspect αντίστοιχα. Σε αυτά τα παράθυρα, κατά την επιλογή ενός αντικειμένου θα εμφανίζονται οι παράμετροι που αφορούν το κάθε aspect, δίνοντας τη δυνατότητα στον χρήστη να μεταβάλλει τις τιμές τους. Για παράδειγμα, στο semanticaspect μπορεί ο χρήστης να προσθαφαιρεί symbols σε ένα αντικείμενο, ενώ στο accessmodel μπορεί να προσθαφαιρεί accesspoints, να τα τροποποιεί και να προσθαφαιρεί functions σε αυτά. Μάλιστα, όσον αφορά τις functions, θα μπορεί να επιλέξει μέσα από την build-in λειτουργικότητα (classes) που προσφέρει η αναπαράσταση Reve.

Άλλη μια δυνατότητα που θέλουμε να καλύπτει η εφαρμογή είναι να προσφέρει στο χρήστη διαφορετικές γωνίες παρατήρησης του κόσμου, δηλαδή να μπορεί να μεταβάλλει την κάμερα παρατήρησης, βλέποντας είτε κάτωψη (άξονες  $xy$ ), είτε πρόσοψη ( $xz$ ) είτε προοπτική ( $xyz$ ). Επίσης, θέλουμε ο χρήστης να έχει τη δυνατότητα να προσαρμόζει το περιβάλλον εργασίας του όπως τον εξυπηρετεί, δηλαδή να μπορεί να μετακινεί και να τοποθετεί αλλού τα διαφορετικά παράθυρα (tabs) που υπάρχουν ταυτόχρονα στην εφαρμογή ή να τα κρύβει και να τα εμφανίζει όπως επιθυμεί.

Επιπλέον, πρέπει όλες οι ενέργειες να είναι προσβάσιμες από το χρήστη και από ένα κεντρικό «Μενού» όπου θα μπορεί να τις αναζητήσει καθώς να δίνεται και η επιλογή της «Βοήθειας», όπου θα εμφανίζονται οδηγίες για τον τρόπο λειτουργίας της εφαρμογής.

Τέλος, θέλουμε ιδανικά να μπορεί ο κόσμος να γίνεται *gendering* και να φορτώνεται απευθείας στην πλατφόρμα ReveWorlds για περαιτέρω επεξεργασία ή σύνδεση με τους εικονικούς πράκτορες.

## 11.2 Τεχνολογίες Υλοποίησης

Σε αυτό το κεφάλαιο παρουσιάζουμε τις τεχνικές επιλογές που κάναμε για να ικανοποιήσουμε στον μέγιστο βαθμό την ανάλυση απαιτήσεων.

Η εφαρμογή αναπτύχθηκε στη γλώσσα προγραμματισμού Java και πιο συγκεκριμένα χρησιμοποιήθηκε το περιβάλλον JavaSE (StandardEdition) Environment 7 της Oracle και το JavaDevelopmentKit (JDK) 7. Η συγκεκριμένη έκδοση προτιμήθηκε γιατί ήταν η πιο πρόσφατη

και ανανεωμένη από άποψη βιβλιοθηκών και λειτουργικότητας κατά την εκκίνηση υλοποίησης της μεταπτυχιακής μας διατριβής. Αναφορικά με την επιλογή της γλώσσας java έναντι άλλης, λάβαμε υπόψη αρκετούς παράγοντες, όπως εξηγούνται παρακάτω.

Πρώτον, η java είναι μια έγκυρη και αξιόπιστη γλώσσα προγραμματισμού. Τα πλεονεκτήματά της είναι η ελεύθερη διάθεσή της (από την εταιρεία που τη σχεδίασε και οδηγεί την ανάπτυξή της, τη Sun), η μεταφερσιμότητα (υπάρχουν εκδόσεις της εικονικής μηχανής – JVM – για πολλά συστήματα), η ευκολία ανάπτυξης και συντήρησης του κώδικα, η ισχυρή standard βιβλιοθήκη της, αλλά και τα πολλά διαθέσιμα frameworks που διατίθενται στον παγκόσμιο ιστό. Επιπλέον, η απόδοση της γλώσσας είναι πολύ καλή, και αποτελεί δημοφιλή επιλογή για web-based λύσεις, χάρις τον ενσωματωμένο μηχανισμό ασφάλειας που διαθέτει. Συνεπώς, η java ικανοποιεί την απαίτηση η εφαρμογή μας να υποστηρίζεται από διαφορετικά λειτουργικά συστήματα και να είναι εύκολα προσβάσιμη στους χρήστες από τον προσωπικό τους υπολογιστή.

Δεύτερον, η java είναι μια αντικειμενοστραφής γλώσσα (object-oriented) που επιτρέπει την επαναχρησιμοποίηση κώδικα και την αναπαράσταση αυτόνομων οντοτήτων με διακριτές λειτουργίες μέσω της δημιουργίας αντικειμένων, γεγονός που μας εξυπηρετεί για την αναπαράσταση των στοιχείων που αποτελούν τον κόσμο μας, όπως το item και τα διάφορα μοντέλα λειτουργίας του.

Ένα πλεονέκτημα ακόμα είναι ότι η java έχει έτοιμες βιβλιοθήκες αναπαράστασης γραφικών GUI, επομένως είναι μια γλώσσα κατάλληλη για εφαρμογές που χρειάζονται γραφική διεπαφή για αλληλεπίδραση με τον χρήστη. Η Java3D ουσιαστικά αποτελεί ένα module επέκτασης των λειτουργιών της Java. Αρκετά ενδιαφέρουσα είναι η δυνατότητα επιλογής συγκεκριμένου τρόπου απεικόνισης της τρισδιάστατης σκηνής, επιτρέποντας έτσι τον έλεγχο της απόδοσης του συστήματος. Η τεχνολογία παρουσιάζει αυξημένες δυνατότητες αλληλεπίδρασης με το χρήστη. Σημαντική είναι και η συνεργασία με άλλα πρότυπα και τεχνολογίες τρισδιάστατης δικτυακής απεικόνισης. Επίσης η εκμετάλλευση των βιβλιοθηκών Java επεκτείνει κατά πολύ το εύρος των δυνατών εφαρμογών και των λειτουργιών που μπορεί να παρέχει.

Το πρότυπο VRML στην αρχή και X3D στη συνέχεια είναι δύο καθαρά δικτυακά πρότυπα. Προδιαγράφουν τη σύνταξη και τη δομή των αρχείων τους. Το VRML πρότυπο έδωσε το έναυσμα για την ευρύτερη κινητοποίηση στο χώρο των τρισδιάστατων δικτυακών τεχνολογιών και περιλαμβάνει στοιχεία και δυνατότητες που απευθύνονται σε μεγάλο εύρος εφαρμογών. Το X3D, ως συνεχιστής της VRML, επεκτείνει αυτές τις δυνατότητες, συμβαδίζει πλήρως με XML και κυρίως προδιαγράφει και τα απαραίτητα χαρακτηριστικά των X3D browsers (Scene Authoring Interface) οριοθετώντας καλύτερα τις όποιες αλληλεπιδράσεις μεταξύ των διάφορων εφαρμογών.

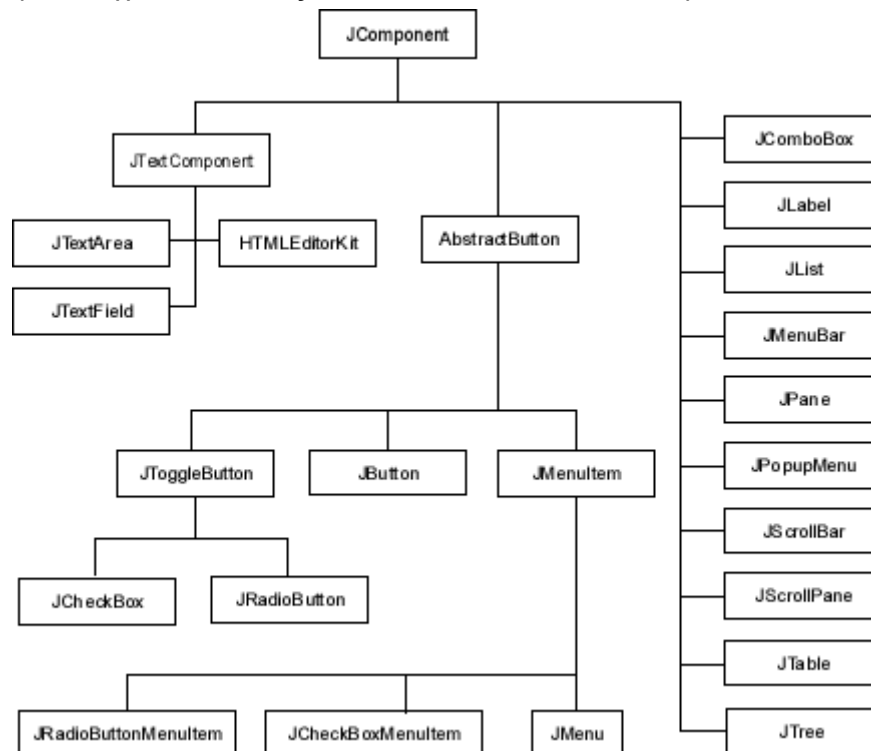
Επιπλέον προσφέρονται δυο πακέτα εργαλείων (toolkit) αναπαράστασης γραφικών, το AbstractWindowToolkit (AWT) και την Swing, που είναι μέρος των JavaFoundationClasses (JFC).

Η AWT είναι μια φορητή βιβλιοθήκη που συνδέει μια εφαρμογή με το GUI της μηχανής και προσφέρει μια υψηλού επιπέδου αφαίρεση αναπαράστασης γραφικών, καθώς κρύβει τις λεπτομέρειες του UI πάνω στο οποίο θα τρέξει η εφαρμογή. Παρέχει μια ποικιλία έτοιμων υλοποιήσεων όσον αφορά συστατικά στοιχεία διεπαφής χρήστη και ένα ικανό μοντέλο χειρισμού γεγονότων (eventhandlingmodel). Επίσης, προσφέρει εργαλεία γραφικών και απεικόνισης, όπως σχήματα, χρώματα, κλάσεις γραμματοσειρών, καθώς και διαφορετικούς τρόπους ταξινόμησης παραθύρων, ανεξαρτήτως ανάλυσης και μεγέθους οθόνης. Επιπλέον, υποστηρίζει την μεταφορά δεδομένων, ώστε να είναι δυνατή η αποκοπή και η επικόλληση ενός αντικειμένου μεταξύ διαφορετικών στοιχείων της εφαρμογής. Η AWT χρησιμοποιείται κυρίως για stand-alone εφαρμογές και applets.

Η βιβλιοθήκη που χρησιμοποιήσαμε εμείς για την εφαρμογή μας είναι η Swing (javax.swing.\*), καθώς έχει όλα τα χαρακτηριστικά της AWT προσφέροντας, ωστόσο, ακόμα περισσότερα ή βελτιστοποιημένα στοιχεία αναπαράστασης, όπως είναι η δένδρική δομή, οι πίνακες, οι λίστες, τα κουμπιά, τα πλαίσια και τα παράθυρα με εμφωλευμένες καρτέλες, ενώ υποστηρίζει και πληθώρα λειτουργικότητας επί αυτών, όπως η δυνατότητα να ταξινομήσεις τα

επιμέρους στοιχεία ή να τα σύρει στην οθόνη (draganddrop).Ακόμα, έχει έτοιμες κλάσεις που υποστηρίζουν την λειτουργικότητα για ενέργειες undo-redo. Επιπλέον, η swing υποστηρίζει διαφορετική αισθητική αναπαράσταση (lookandfeel) της εφαρμογής σε αντίθεση με την AWT, πράγμα που την καθιστά προσαρμόσιμη στα γούστα του χρήστη. Για παράδειγμα, μια εφαρμογή μπορεί να έχει την default αισθητική μιας Java εφαρμογής ή την αισθητική των windows ή των mac. Ακόμα, υποστηρίζει επιπλέον δυνατότητες, όπως είναι η χρήση εικονιδίων και pop-up menus που δεν τα παρέχει η AWT. Τέλος, σε αντίθεση με την AWT που τα στοιχεία της εξαρτώνται από τα αντίστοιχά τους σε κώδικα μηχανής, η Swing βιβλιοθήκη είναι εξολοκλήρου σχεδιασμένη σε java, γεγονός που την καθιστά πλήρως αυτόνομη από τους περιορισμούς της συγκεκριμένης πλατφόρμας υλοποίησης.

Όπως γίνεται αντιληπτό, η πληθώρα στοιχείων διεπαφής που προσφέρει η Swing παράλληλα με την υποστήριξη της λειτουργικότητας των συστατικών της AWT βιβλιοθήκης και την ανεξαρτησία της από το λειτουργικό σύστημα, την καθιστά ιδανική για την εφαρμογή μας, προσφέροντας τη δυνατότητα επέκτασης και εμπλουτισμού των υπάρχοντων επιλογών για την διευκόλυνση χειρισμού της από υποψήφιους χρήστες. Αυτό δεν σημαίνει ότι δεν χρησιμοποιούμε στοιχεία από την AWT – άλλωστε η Swing έχει φτιαχτεί με βάση αυτήν, επομένως κάτι τέτοιο θα ήταν σχεδιαστικά αδύνατο για ορισμένα συστατικά – αλλά σημαίνει πως εκμεταλλευόμαστε τις δυνατότητες της Swing, αποφεύγοντας κλάσεις της AWT όπου δύναται χωρίς, όμως, να συνδυάζουμε συστατικά στοιχεία από τις δυο βιβλιοθήκες με λάθος τρόπο, πχ τοποθετώντας ένα JButton σε ένα Frame αντί για ένα JFrame.



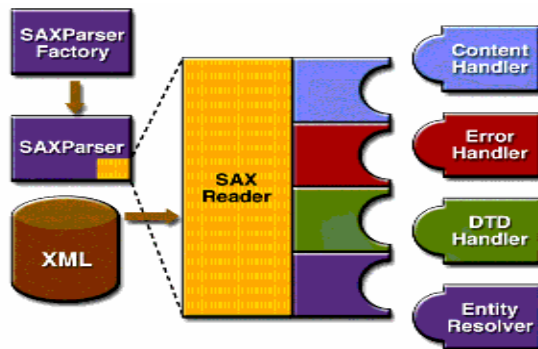
Εικόνα 64: Τα στοιχεία της Swing από το JComponent

Πέραν της ευκολίας αναπαράστασης γραφικών που έχουν οι παραπάνω βιβλιοθήκες της java, έχουν κλάσεις και για τον χειρισμό γεγονότων (eventhandling) μέσω της διεπαφής, όπως αναφέρθηκε. Η λειτουργικότητα αυτή είναι κυρίαρχη στην εφαρμογή μας, εφόσον πρέπει το πρόγραμμα να αντιλαμβάνεται τις ενέργειες του χρήστη και να αντιδρά κατάλληλα, επιτρέποντας ή όχι μια ενέργεια υπό συνθήκες. Για παράδειγμα, είναι ουσιαστική η δυνατότητα αναγνώρισης εισόδου ενός αντικειμένου στον κόσμο από τις επιλογές που έχει κάνει ο χρήστης, καθώς η σύνθεση του εικονικού κόσμου πραγματοποιείται καθαρά από τον ίδιο και δεν αποτελεί μια λειτουργία που εκτελείται αυτόματα. Επίσης, η αναγκαιότητα των eventhandlers και των listeners γίνεται πιο εύκολα κατανοητή αν αναλογιστούμε ότι μια επιλογή μέσω της διεπαφής

χρήστη, όπως είναι το πάτημα ενός κουμπιού, δεν σημαίνει τίποτα από μόνη της αν δεν υπάρχει κώδικας που να ορίζει συγκεκριμένα την λειτουργία μιας ενέργειας, καθώς και τις προϋποθέσεις υπό τις οποίες αυτή μπορεί να εκτελεστεί.

Μια επιπλέον δυνατότητα που μας προσφέρει η java και είναι απαραίτητη στην εφαρμογή μας είναι το πακέτο κλάσεων για επεξεργασία αρχείων (java.io.\*), δεδομένου ότι χρειαζόμαστε αρχεία τόσο για να φορτώσουμε ένα αντικείμενο στον κόσμο όσο και για να δημιουργήσουμε την αναπαράσταση του κόσμου ή να φορτώσουμε έναν έτοιμο κόσμο για περαιτέρω επεξεργασία στην εφαρμογή. Συνδυαστικά με την input/output βιβλιοθήκη, η java παρέχει μια επιπλέον βιβλιοθήκη που αναγνωρίζει, διαβάζει, επεξεργάζεται και δημιουργεί αρχεία σε μορφή xml, γνωστή ως JAXP (Java API for XML Processing). Δεδομένου ότι η αναπαράσταση REVE χρησιμοποιεί την xml-based γλώσσα ve1 για τον εικονικό κόσμο, η εν λόγω βιβλιοθήκη της java κρίνεται απαραίτητο στοιχείο για την αναπαράσταση του εικονικού μας κόσμου.

Η JAXP βιβλιοθήκη αποτελείται από διάφορα πακέτα. Αξιοποιεί τις προδιαγραφές ενός προγράμματος ανάλυσης (parser) XML με το Simple API for XML Parsing (SAX) και το Document Object Model (DOM), επιτρέποντας κατά αυτό τον τρόπο την μεταφορά/μετάφραση των δεδομένων σαν μια ροή γεγονότων ή την αναπαράσταση τους μέσω αντικειμένου. Επιπλέον, υποστηρίζει το πρότυπο XSLT (Extensible Stylesheet Language Transformations), μέσω του οποίου δίνει τον έλεγχο στον προγραμματιστή για την αναπαράσταση των δεδομένων που του επιτρέπει να τα μετατρέψει σε άλλα xml αρχεία ή αρχεία άλλου τύπου, όπως η html. Επίσης, η JAXP επιτρέπει στον προγραμματιστή να δουλέψει με DTDs (Document Type Definition) για να αποφύγει συγκρούσεις μεταξύ ονομάτων, δεδομένου ότι το DTD είναι ένα έγγραφο που περιγράφει την εσωτερική δομή ενός αρχείου xml, ορίζοντας τα επιτρεπτά στοιχεία (elements) και χαρακτηριστικά (attributes) του. Αυτό για την δική μας περίπτωση είναι ιδιαίτερα σημαντικό καθώς και η ve1 χρησιμοποιεί το δικό της DTD, το οποίο μπορούμε να αξιοποιήσουμε. Επίσης, σχεδιασμένη για να είναι ευέλικτη, η JAXP επιτρέπει τη χρήση οποιουδήποτε προγράμματος ανάλυσης συμβατού με XML μέσα από την εφαρμογή. Αυτό επιτυγχάνεται με αυτό που ονομάζεται ένα «pluggability» στρώμα, που επιτρέπει τη σύνδεση του SAX ή του DOM API με την εφαρμογή. Το στρώμα αυτό επιτρέπει επίσης τη σύνδεση ενός επεξεργαστή XSL, επιτρέποντάς τον έλεγχο πάνω στον τρόπο εμφάνισης των δεδομένων XML.



Εικόνα 65: Δομή JAXP

Τέλος, η java έχει ειδικές βιβλιοθήκες για την αναπαράσταση αντικειμένων VRML και X3D. Πιο συγκεκριμένα, στην εφαρμογή μας χρησιμοποιούμε την XJ3D, που πρόκειται για ένα έργο του Web3D Consortium που εστιάζει στη δημιουργία ενός toolkit για την αναπαράσταση περιεχομένου σε VRML και X3D και είναι εξολοκλήρου υλοποιημένο σε java. Εξυπηρετεί ένα διπλό σκοπό, ο πρώτος αφορά την ύπαρξη μιας βάσης κώδικα για τον πειραματισμό σε νέες περιοχές των προδιαγραφών X3D και το δεύτερο αφορά σε βιβλιοθήκη που ενθαρρύνει τους προγραμματιστές να τη χρησιμοποιούν στις εφαρμογές τους για την υποστήριξη της τεχνολογίας X3D. Χρησιμοποιείται, λοιπόν, για να εισάγεις vtml περιεχόμενο σε μια εφαρμογή ή



για να δημιουργήσεις ένα πλήρες πρόγραμμα περιήγησης (browser), που εμφανίζει τα 3d πρότυπα μοντελοποίησης για τους τύπους VRML97 και X3D.

Ένα κλασικό πρόγραμμα περιήγησης –xj3dbrowser– έχει την μορφή που παρουσιάζεται στην Εικόνα 66 και επιτρέπει τις παρακάτω λειτουργίες:



**Open:** Εμφανίζει ξεχωριστό παράθυρο που επιτρέπει στον χρήστη να περιηγηθεί στο σύστημα φακέλων του υπολογιστή του και να επιλέξει ένα VRML97 ή X3D αρχείο, το οποίο εμφανίζεται την κεντρική οθόνη.

Η γραμμή **Location** εμφανίζει την ακριβή τοποθεσία του αρχείου στον υπολογιστή και κρατάει ιστορικότητα των αντικειμένων που έχουν φορτωθεί. Επομένως, επιλέγοντας ένα άλλο path στο Location από τα διαθέσιμα και πατώντας Go, φορτώνεται το αντικείμενο που ορίζεται στο path.



**Reload:** Επαναφορτώνει το αρχείο που εμφανίζεται στην κεντρική οθόνη ή στο Location αντίστοιχα.



**Fly:** Επιτρέπει στο χρήστη να περιηγηθεί στο χώρο σαν να ίππεται, σαν να αιωρείται πάνω από το αντικείμενο που εξετάζει.



**Pan:** Επιτρέπει στον χρήστη να περιεργαστεί το αντικείμενο κατά μήκος των αξόνων x και y.



**Tilt:** Επιτρέπει στο χρήστη να προσκλίνει στο αντικείμενο.



**Walk:** Επιτρέπει στο χρήστη να περιηγηθεί ελεύθερα στο χώρο, πλησιάζοντας ένα αντικείμενο ή απομακρυνόμενος από αυτό.



**Track:** Επιτρέπει στο χρήστη να περιεργαστεί το αντικείμενο (ή τον κόσμο) από όλες τις πλευρές του, περιστρέφοντας το με σημείο αναφοράς τον άξονα κατά μήκος της πλευράς που επιλέγει, αποκτώντας έτσι μια συνολική εικόνα για αυτό.



**Examine:** Επιτρέπει στο χρήστη να επεξεργαστεί σφαιρικά ένα αντικείμενο, περιστρέφοντας το προς κάποιον άξονα που επιλέγει με σημείο αναφοράς το κέντρο του.



**PreviousViewpoint:** Μεταφέρει την οπτική γωνία του αντικείμενου στην επόμενη που ορίζεται, αν ορίζεται.



**NextViewpoint:** Μεταφέρει την οπτική γωνία του αντικείμενου στην προηγούμενη που ορίζεται, αν ορίζεται.



**Return to current Viewpoint:** Μεταφέρει την οπτική γωνία του αντικειμένου στην προεπιλεγμένη που έχει το αντικείμενο κατά την εισαγωγή του στην σκηνή.



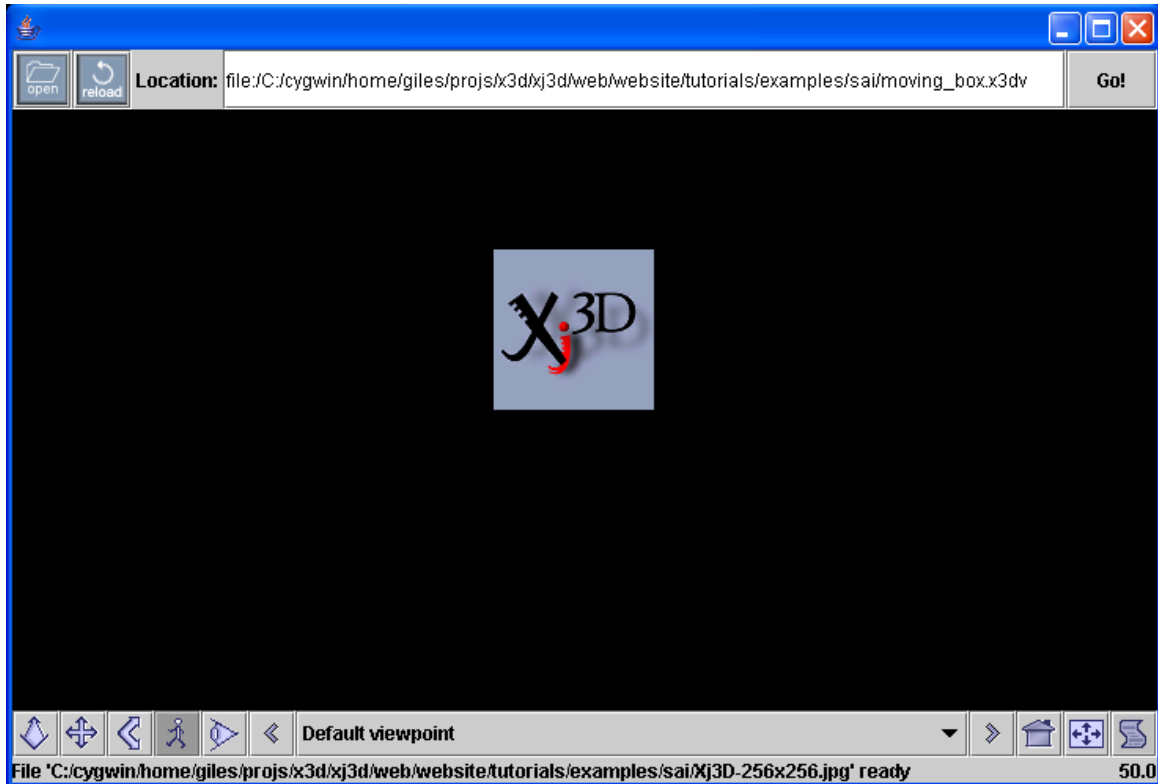
**Look At:** Επιτρέπει στο χρήστη να πατήσει σε ένα σημείο του χώρου και να μεταφερθεί σε αυτό.



**Fit to World:** Μεταφέρει το αντικείμενο στο κέντρο της οθόνης ώστε να καταλαμβάνει όλο τον επιτρεπτό χώρο που υπάρχει στη σκηνή. Κατά αυτόν τον τρόπο, δίνει την αίσθηση του ολικού zoom-in στον χρήστη.

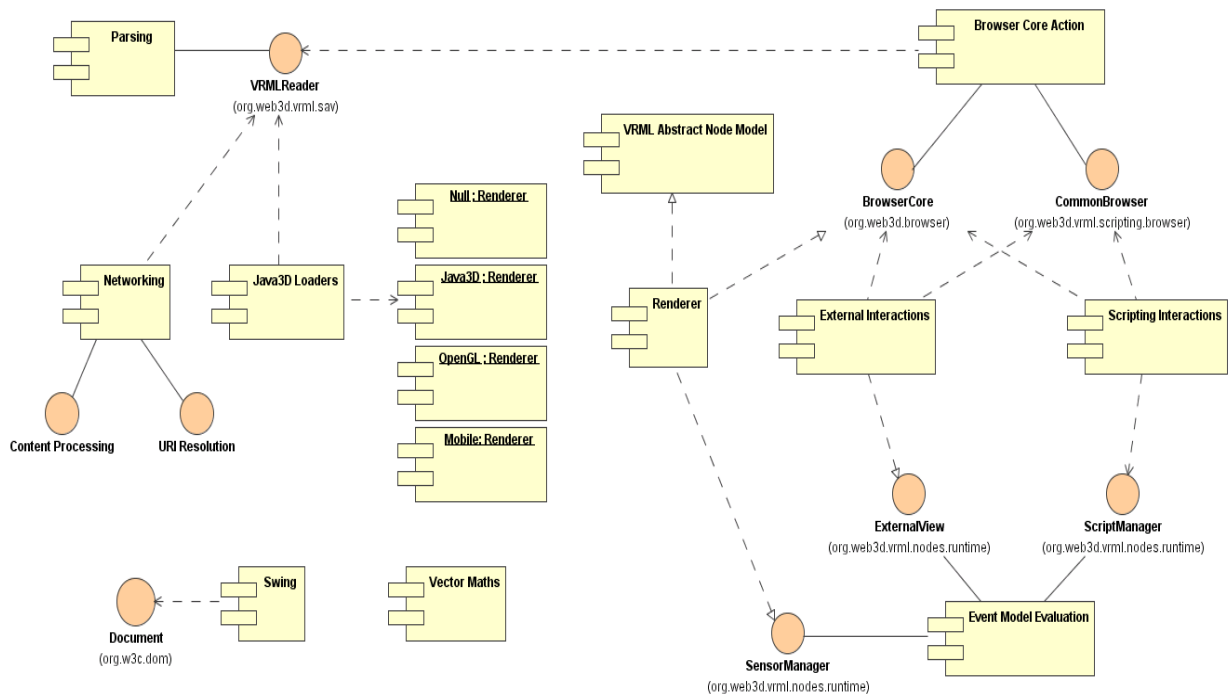


**ShowBrowserConsole:** Ανοίγει ένα παράθυρο με πληροφορίες για τον browser. Κατά κύριο λόγο, εμφανίζει προειδοποιήσεις και πληροφορίες για το αντικείμενο που βρίσκεται στη σκηνή.



Εικόνα 66: Xj3DBrowser

Καθίσταται σαφές ότι η XJ3D είναι απαραίτητη για να εμφανίσουμε τα αντικείμενα που θέλουμε να επιλέξουμε για να σχεδιάσουμε τον εικονικό κόσμο, για να τα εισάγουμε στον κόσμο, για να σχεδιάσουμε τον κόσμο και φυσικά για να περιηγηθούμε σε αυτόν. Στην εφαρμογή μας χρησιμοποιήσαμε την πιο πρόσφατη έκδοση της βιβλιοθήκης xj3d, που έχει αριθμό έκδοσης 2.0.



**Εικόνα 67: Αρχιτεκτονική του Xj3D**

Προς διευκόλυνση ανάπτυξης του κώδικα της εφαρμογής χρησιμοποιήσαμε το εργαλείο NetBeans IDE 7.3.1. σε περιβάλλον Windows 7. Το NetBeans IDE (Integrated Development Environment) είναι μια open-source, ελεύθερα προσβάσιμη πλατφόρμα ανάπτυξης λογισμικού γραμμένη σε Java, που επιτρέπει στις εφαρμογές να αναπτυχθούν από ένα σύνολο αρθρωτών στοιχείων λογισμικού, γνωστά ως modules. Παρέχει πληθώρα εργαλείων που διευκολύνουν τον προγραμματιστή να συνδέσει τα επιμέρους τμήματα που συνθέτουν μια εφαρμογή και χρησιμοποιείται ευρέως για την ανάπτυξη κώδικα σε java, τόσο για ελεύθερη χρήση όσο και επαγγελματικά.

Επίσης, για την ανάγνωση και επεξεργασία νεrl αρχείων, που αναπαριστούν τον εικονικό κόσμο για την πλατφόρμα Reve χρησιμοποιήσαμε την τελευταία έκδοση 6.7.8.2 του εργαλείου Notepad++. Πρόκειται για μια έκδοση που θυμίζει το κλασικό σημειωματάριο (Notepad) των Windows, αλλά στην ουσία είναι ένα πρόγραμμα επεξεργασίας πηγαίου κώδικα που υποστηρίζει πολλές γλώσσες, εκ των οποίων και η xml. Πέραν τούτου, για την επαλήθευση της σωστής σύνταξης της νεrl και της αναπαράστασης του εικονικού κόσμου χρησιμοποιήσαμε και την πλατφόρμα ReveWorlds.

Τέλος, για την πρωταρχική επεξεργασία και την απεικόνιση των 3D αντικειμένων που είτε δημιουργούσαμε είτε τα παίρναμε έτοιμα, ώστε να πληρούν τις προδιαγραφές της αναπαράστασης REVE, χρησιμοποιήσαμε το εργαλείο VrmlPad 3.0 και το Cortona 3DViewer. Το VrmlPad είναι εργαλείο ανάπτυξης και επεξεργασίας κώδικα VRML, κατάλληλο για την σχεδίαση τρισδιάστατων αντικειμένων. Από την άλλη, το Cortona 3D Viewer (παλαιότερα γνωστό ως Cortona VRML Client) λειτουργεί ως ένα VRML plug-in για τα πιο δημοφιλή προγράμματα περιήγησης στο Διαδίκτυο (Internet Explorer , Mozilla Firefox , Google Chrome , Opera) και εφαρμογών γραφείου (Microsoft PowerPoint , Microsoft Word). Υποστηριζόμενο από τα MS Windows το Cortona 3D Viewer είναι ελεύθερο για προσωπική και ακαδημαϊκή χρήση και δίνει την δυνατότητα φόρτωσης ενός .wrl αρχείου για την απεικόνιση του περιεχομένου του στον browser του χρήστη.

## **Κεφάλαιο 12 - Υλοποίηση Εφαρμογής Για Την Επιλογή Τρισδιάστατων Αντικειμένων Για Τη Σύνθεση Εικονικών Κόσμων**

Σε αυτό το κεφάλαιο παρουσιάζουμε τα επιμέρους τμήματα που συνθέτουν την εφαρμογή, τη λειτουργικότητα που έχουν και πώς αλληλοσυνδέονται μεταξύ τους, εστιάζοντας στο κομμάτι της επιλογής των τρισδιάστατων αντικειμένων για την σύνθεση του εικονικού κόσμου.

Δεδομένου ότι η εφαρμογή συμπύχθηκε από δυο κομμάτια το πρώτο κομμάτι, που αναλύεται στην συγκεκριμένη διπλωματική διατριβή, αφορά την επιλογή και ανάλυση της δομής των τρισδιάστατων αντικειμένων για τη σύνθεση εικονικών κόσμων ενώ το δεύτερο κομμάτι το οποίο αναπτύχθηκε και περιγράφεται αναλυτικά στην διπλωματική διατριβή της Έντα Τσελεμέγκου με τίτλο «Εφαρμογή Reve World Designer - Σχεδίαση εικονικών κόσμων για την πλατφόρμα REVE» αφορά την σχεδίαση του εικονικού κόσμου, δηλαδή τη φυσική αναπαράσταση του, μετατρέποντας τα αντικείμενα που λαμβάνει από το πρώτο κομμάτι σε items που συνθέτουν τον εικονικό κόσμο, βάσει των προδιαγραφών της αναπαράστασης REVE.

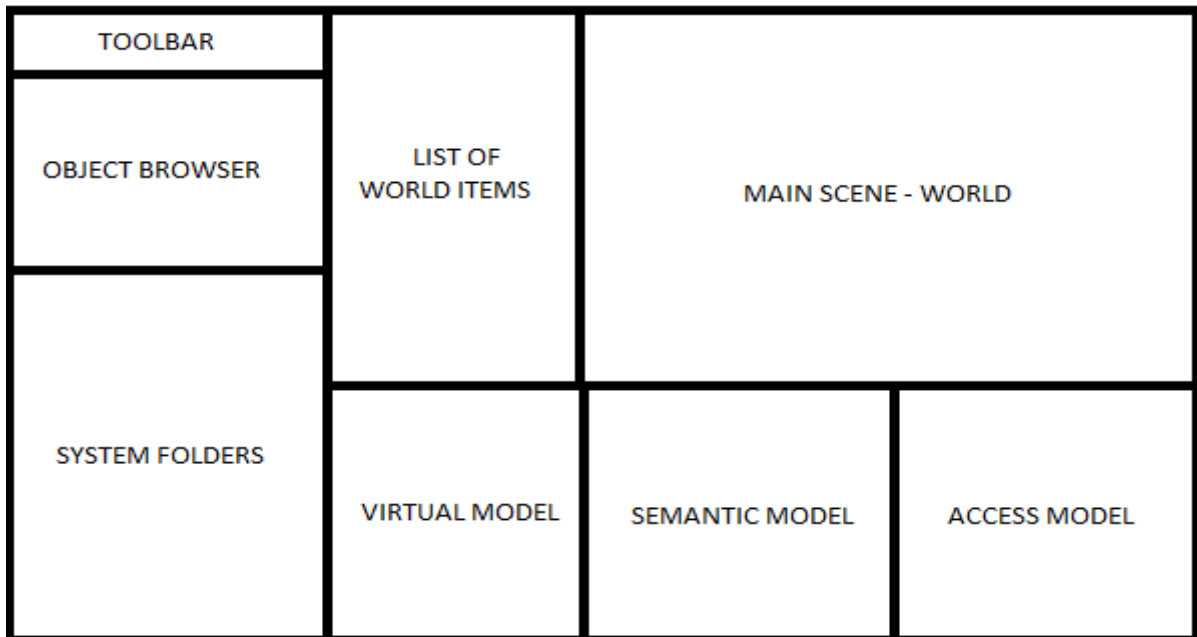
### **12.1 Δομή Εφαρμογής**

Η παρούσα διπλωματική διατριβή σκοπό είχε μέσω της επιλογής ενός τρισδιάστατου αντικειμένου να μπορεί να επιτευχθεί η σύνθεση εικονικών κόσμων και πιο συγκεκριμένα η συγκεκριμένη διπλωματική διατριβή αφορά την επιλογή αντικειμένου, βάσει των προδιαγραφών που θέτει η αναπαράσταση Reve, για τη χρήση ολόκληρου του αντικειμένου ή συγκεκριμένου κομματιού του για την δημιουργία του εικονικού κόσμου που θέλουμε να δημιουργήσουμε.

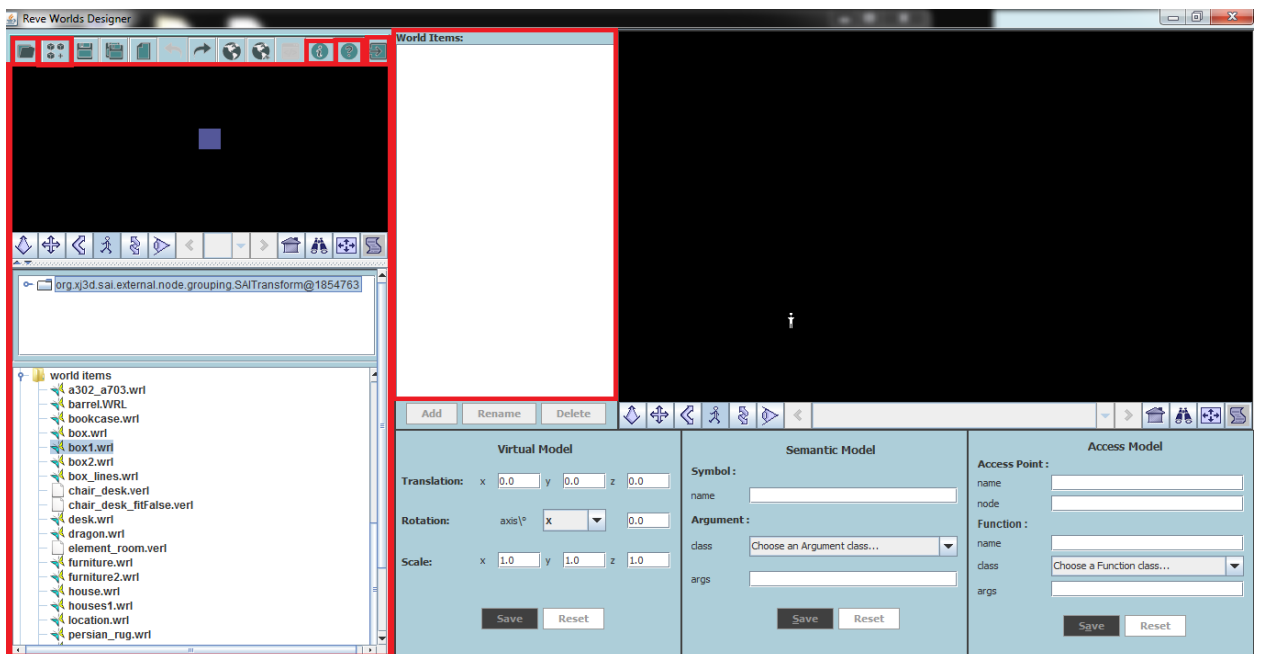
Το δεύτερο κομμάτι το οποίο και περιγράφεται αναλυτικά στην διπλωματική διατριβή της Έντα Τσελεμέγκου με τίτλο «Εφαρμογή Reve World Designer - Σχεδίαση εικονικών κόσμων για την πλατφόρμα REVE» αφορά την πλήρη σχεδίαση του εικονικού κόσμου και την ολοκλήρωση της συγκεκριμένης πλατφόρμας. Αφορά δηλαδή τη φυσική αναπαράσταση του κόσμου, μετατρέποντας τα αντικείμενα που λαμβάνει από το πρώτο κομμάτι σε items που συνθέτουν τον εικονικό κόσμο, βάσει των προδιαγραφών της αναπαράστασης REVE.

Σε πρώτο στάδιο συμφωνήθηκε να φτιαχτεί το σχέδιο της γραφικής διεπαφής ως ένα mock-up των παραθύρων από τα οποία θα αποτελείται. Το πρώτο κομμάτι, το οποίο αναλύεται και στην συγκεκριμένη μεταπτυχιακή διατριβή, χωρικά είναι το αριστερό κομμάτι και αφορά στην επιλογή του τρισδιάστατου αντικειμένου ή κάποιου συγκεκριμένου κομματιού του αντικειμένου το οποίο επιθυμεί ο χρήστης να εισάγει στον εικονικό κόσμο. Το δεύτερο κομμάτι αφορά τη σχεδίαση του εικονικού κόσμου, τροποποιώντας κάποια στοιχεία του πρώτου τμήματος και εισάγοντας καινούρια. Σε αυτή την διπλωματική διατριβή θα αναλυθούν τα κομμάτια που αφορούν, από αριστερά, την γραμμή εργαλείων (Toolbar), την απεικόνιση των αντικειμένων (Object Browser), τον κατάλογο των αρχείων του χρήστη (System Folders) και την λίστα των αντικειμένων που εισάγει ο χρήστης στον κόσμο (List of World Items). Επιπλέον θα αναλυθεί και κατά την επιλογή ενός αντικειμένου το πώς δημιουργείται το γράφημα σκηνής του αλλά και το πώς μπορεί να επιλεγεί και να αποδοθεί ένα μόνο κομμάτι από όλο το αντικείμενο.

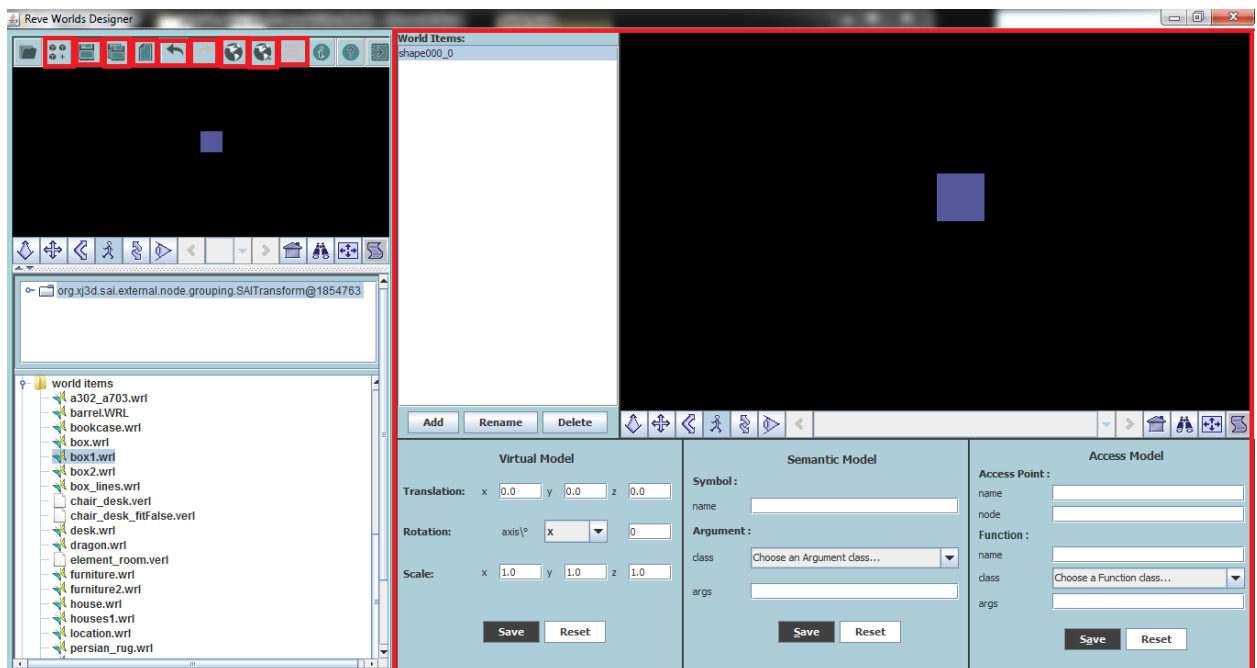
Στην παρακάτω εικόνα φαίνεται η σχεδιαστική δομή που συμφωνήθηκε να έχει η εφαρμογή. Με το πέρας των δύο διπλωματικών διατριβών και την συγχώνευση τους το τελικό αποτέλεσμα θα έχει την παρακάτω μορφή.



Εικόνα 68: Σχεδιαστική δομή παραθύρων



Εικόνα 69: Πρώτο κομμάτι – επιλογή τρισδιάστατου αντικειμένου



Εικόνα 70: Δεύτερο κομμάτι – σχεδίαση εικονικού κόσμου

Παρακάτω παρουσιάζουμε τα συστατικά που αποτελούν την εφαρμογή μας, την λειτουργικότητά τους, καθώς και τις δυσκολίες που αντιμετωπίσαμε κατά την υλοποίηση. Η ανάλυση της εφαρμογής θα γίνει από τα στοιχεία που συνθέτουν την γραμμή εργαλείων, μετά θα αναλυθεί το παράθυρο που απεικονίζει τον κατάλογο των αρχείων του χρήστη, ύστερα θα αναλυθεί το παράθυρο που είναι υπεύθυνο για την απεικόνιση των αντικειμένων (Object Browser) και παράλληλα το παράθυρο που εμφανίζει το γράφημα σκηνης ενός απεικονισμένου αντικειμένου. Τέλος θα γίνει μια αναφορά στην λίστα των αντικειμένων που εισάγει ο χρήστης στον κόσμο. Στην συγκεκριμένη διπλωματική διατριβή η τελική εισαγωγή του αντικειμένου δεν μπορεί να απεικονιστεί με απόλυτη σαφήνεια, μπορεί όμως να δοθεί ένα τελικό αποτέλεσμα τον αντικειμένων που επιλέγονται. Στην διπλωματική διατριβή της συναδέρφου Εντα Τσελεμέγκου παντρεύεται αυτό το κομμάτι και φαίνεται το πώς απεικονίζονται τα αντικείμενα μετά την εισαγωγή τους στο κεντρικό παράθυρο της εφαρμογής.


## 12.2 Η γραμμή Εργαλείων (Κεντρικό Μενού)






Εικόνα 71: Γραμμή εργαλείων

Η γραμμή εργαλείων ή το κεντρικό μενού που αφορά την συγκεκριμένη διπλωματική διατριβή

περιλαμβάνει τις επιλογές του «OpenFile»  το οποίο χρησιμοποιείται για την εύρεση και το


άνοιγμα των τρισδιάστατων αρχείων, του «AddNodetoWorld»  όπου η λειτουργικότητα του χρησιμεύει στην εισαγωγή ενός αντικειμένου ή ενός κομματιού από ένα τρισδιάστατο

αντικείμενο στον εικονικό μας κόσμο, στην επιλογή «About Us» , το οποίο εμφανίζει τα στοιχεία των δημιουργών της εφαρμογής και των επιβλεπόντων καθηγητών, την επιλογή

«Help» , η οποία στοχεύει να κατατοπίσει τον χρήστη με πληροφορίες και οδηγίες χρήσης της εφαρμογής και τέλος της επιλογής «Exit without saving»  η οποία είναι υπεύθυνη για τον τερματισμό της εφαρμογής. Παρακάτω θα γίνει μια εκτενής ανάλυση σε κάθε στοιχείου της γραμμής εργαλείων αναλυτικά.

### 12.2.1 Εύρεση Και Άνοιγμα Τρισδιάστατων Αντικειμένων Με Την Επιλογή «OpenFile»

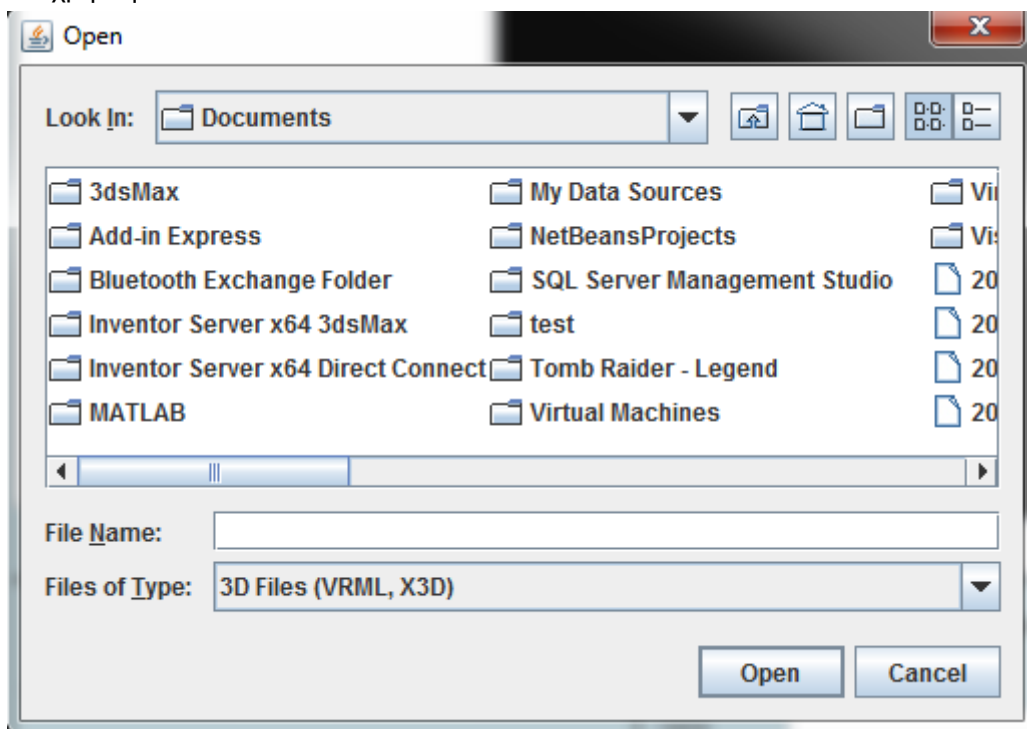
Το πρώτο βήμα για την σχεδίαση ενός εικονικού κόσμου είναι η επιλογή των αντικειμένων που θα τον αποτελούν. Η αναζήτησή και η εισαγωγή τους πρέπει να είναι μια εύκολη και γρήγορη διαδικασία για τον χρήστη. Αποφασίστηκε η εισαγωγή των αντικειμένων να μπορεί να γίνει με δυο τρόπους για μεγαλύτερη πληρότητα της εφαρμογής αλλά και για μεγαλύτερη ευκολία του χρήστη. Ο πρώτος τρόπος, που είναι και ο πιο κλασικός, επιτυγχάνεται από το κεντρικό μενού,

από την επιλογή OpenFile .



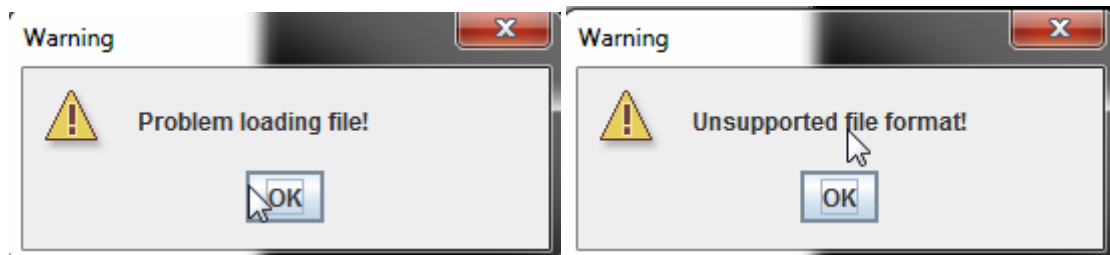
Εικόνα 72: Γραμμή εργαλείων- Επιλογή OpenFile

Για την επιλογή «OpenFile» αξιοποιώντας το JFileChoosercomponent της swing, δίνουμε τη δυνατότητα στον χρήστη να περιηγηθεί στους φακέλους του υπολογιστή του, με default τον φάκελο MyDocuments, για να επιλέξει τα αρχεία με κατάληξη .wrl (για VRML97 αντικείμενα) ή .x3dn (για X3D αντικείμενα), όπως φαίνεται στην Εικόνα 73. Ο δεύτερος τρόπος ο οποίος θα αναλυθεί πιο μετά αφορά την εύρεση των αντικειμένων μέσα από τον κατάλογο των αρχείων του χρήστη.



Εικόνα 73: OpenFile

Σε περίπτωση που ο χρήστης επιλέξει να βλέπει όλα τα αρχεία (AllFiles) στο Files of Type και διαλέξει ένα αρχείο μη συμβατό με την εφαρμογή, δηλαδή ούτε .wrl ούτε .x3dn, του εμφανίζεται ένα σχετικό μήνυμα λάθους (Εικόνα 74).



Εικόνα 74: Προσπάθεια ανοίγματος μη συμβατού αρχείου από το «OpenFile»

Στην περίπτωση, όμως, που το αρχείο που διαλέξει είναι αρχείο κόσμου ver1, εμφανίζεται σε ξεχωριστό παράθυρο ο κώδικας του αρχείου (Εικόνα 75). Η δυνατότητα αυτή δίνεται στο χρήστη στα πλαίσια αναζήτησης αντικειμένου, εφόσον μέσα από τον κώδικα της ver1, μπορεί να εντοπίσει ένα αντικείμενο που τον ενδιαφέρει.



Εικόνα 75: Εμφάνιση κώδικα των ver1 αρχείων

### 12.2.2 Εισαγωγή Ενός Αντικειμένου Με Την Επιλογή «AddNodetoWorld»

Έχοντας ανοίξει ένα αρχείο, η προσθήκη ενός αντικειμένου στην σκηνή μπορεί να γίνει με δύο διαφορετικούς τρόπους. Ο πρώτος τρόπος επιτυγχάνεται μέσω του κεντρικού μενού με το



κουμπί «AddNodetoWorld» (Εικόνα 76). Ο δεύτερος τρόπος θα αναλυθεί πιο κάτω και αφορά την εισαγωγή του αντικειμένου μέσω του γραφήματος σκηνής. Ο δεύτερος τρόπος γίνεται μέσω της χρήσης του ποντικιού και πιο συγκεκριμένα με την χρήση του δεξιού κλικ.





**Εικόνα 76: Γραμμή εργαλείων- Επιλογή AddNodetoWorld**

Έχοντας πραγματοποιηθεί η επιλογή του κόμβου που πρόκειται να εισαχθεί, και εφόσον ο κόμβος πληρεί της προϋποθέσεις που έχουν οριστεί, τότε επιλέγοντας το κουμπί “AddNodetoWorld” εισάγεται ο κόμβος στο γράφημα σκηνής. Σε κάθε εναλλαγή επιλογής ενός κόμβου ή ενός αντικειμένου ελέγχεται αν ο κόμβος που έχει επιλεγεί είναι DEFTransform. Μόνο σε αυτή την περίπτωση ενεργοποιείται η δυνατότητα προσθήκης του αντικειμένου στο γράφημα σκηνής.

Σε αυτό το σημείο αντιμετωπίσαμε δυο ζητήματα. Το πρώτο αφορούσε την προσθήκη του κόμβου μέσα από το κεντρικό μενού, καθώς αν ο χρήστης περιηγούνταν στο γράφημα σκηνής επιλέγοντας διαφορετικούς κόμβους, οι κόμβοι αυτοί προστίθενται όλοι – και όχι μόνο ο τελευταίος - κατά το πάτημα του AddNodetoWorld, γεγονός που δεν συνέβαινε με το δεξί κλικ. Διαπιστώσαμε πως αυτό οφειλόταν στο γεγονός πως με το κουμπί του κεντρικού μενού δεν ήταν διακριτός ποιος κόμβος έχει επιλεγεί, με αποτέλεσμα σε κάθε επιλογή κόμβου να συσσωρεύονται στη μνήμη και να προστίθενται όλοι. Αυτό λύθηκε με τη χρήση μιας static μεταβλητής που κρατάει πληροφορία για την τελευταία επιλογή και καθαρίζει τα προηγούμενα δεδομένα κάθε φορά, οπότε ο κόμβος που τελικά προστίθεται είναι μόνο ο τελευταίος.

Το δεύτερο ζήτημα αφορούσε την προσπάθεια να επιτρέψουμε οποιονδήποτε Transform κόμβο του γραφήματος σκηνής να προστεθεί στον κόσμο, αυξάνοντας τις επιλογές του χρήστη. Αυτό προϋπέθετε να μετατρέψουμε εμείς τον κόμβο σε DEFnode. Ωστόσο, αυτό δεν κατέστη δυνατό, καθώς παρότι η χρίζεπιτρέπεται να βρεις τους DEF κόμβους με τις μεθόδους `getNamedNodes()/getNamedNode()` δεν προσφέρει κάποια μέθοδο για να ορίσεις έναν κόμβο ως DEF. Επίσης, οι κόμβοι λαμβάνονται από την μνήμη, όπου φορτώνονται, επομένως, ακόμα και αν βρίσκαμε έναν τρόπο να παρέμβουμε εκεί μετατρέποντας έναν Transform κόμβο σε DEF, αυτό θα είχε μόνο προσωρινή λειτουργικότητα, καθώς θα έπρεπε αυτός ο DEF κόμβος να ορίζεται και στο αρχείο, από όπου αντλεί τα `item sources` η `verl`. Αυτό τελικά συνεπάγεται πως ο χρήστης πρέπει χειροκίνητα να ανοίξει τον κώδικα ενός 3dαρχείου και να ονοματίσει ένα Transform του ως DEF αν θέλει να το χρησιμοποιήσει οπωσδήποτε στον κόσμο.

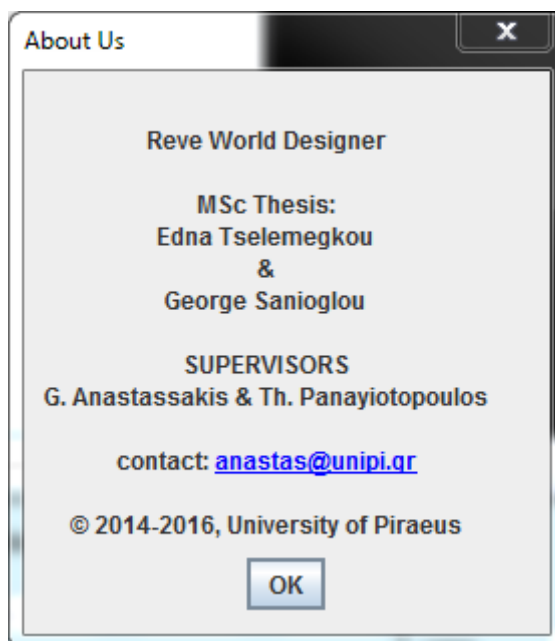
### 12.2.3 Επιλογή «Πληροφορίες Εφαρμογής»

Η γραμμή εργαλείων διαθέτει μία επιλογή που λέγεται «About Us» .




**Εικόνα 77: Γραμμή εργαλείων- Επιλογή «AboutUs»**

Δεν παρέχει κάποια λειτουργικότητα μέσα στην εφαρμογή, απλά έχει ως στόχο να εμφανίζει τα στοιχεία για την δημιουργία της εφαρμογής και τα πνευματικά δικαιώματα, όπως φαίνεται παρακάτω.



Εικόνα 78: Το παράθυρο «About Us»

#### 12.2.4 Επιλογή «Βοήθεια»

Η γραμμή εργαλείων διαθέτει μία επιλογή που λέγεται «Help» , .



Εικόνα 79: Γραμμή εργαλείων- Επιλογή «Help»


Αυτή η επιλογή στοχεύει στο να κατατοπίσει τον χρήστη με πληροφορίες και οδηγίες χρήσης της εφαρμογής. Για αυτό το σκοπό έχουμε ενσωματώσει ένα pdf στο πακέτο της εφαρμογής το οποίο ανοίγει κατά το πάτημα του πλήκτρου «Help». Με την εισαγωγή του χρήστη στην εφαρμογή είναι εφικτή η λείψει βασικών οδηγιών για το πώς δουλεύει η εφαρμογή. Οι οδηγίες που μπορεί να βρει ο χρήστης καλύπτουν όλη την εφαρμογή και πιο συγκεκριμένα στοχεύουν στο να τον κατατοπίσουν στα παρακάτω κεφάλαια:

Τι αρχεία υποστηρίζει η εφαρμογή;

1. Πώς ανοίγω ένα αρχείο;
2. Πώς προσθέτω ένα αντικείμενο στη σκηνή;
3. Πώς φτιάχνω αντίγραφο ενός item;
4. Πώς μετονομάζω ένα item;
5. Πώς διαγράφω ένα item;
6. Πώς αλλάζω το virtual model ενός item;
  - a. Πώς αλλάζω το translation;
  - b. Πώς αλλάζω το rotation;
  - c. Πώς αλλάζω το scale;
7. Πώς αλλάζω το semantic model ενός item;
8. Πώς αλλάζω το access model ενός item;
9. Τι κάνει το undo;
10. Τι κάνει το redo;

11. Πώς αδειάζω τη σκηνή;
12. Πώς αποθηκεύω έναν κόσμο;
13. Πώς βλέπω τον κώδικα ενός αρχείου ver1;
14. Πώς φορτώνω έναν κόσμο στη σκηνή;
15. Πώς προσθέτω έναν κόσμο στη σκηνή;

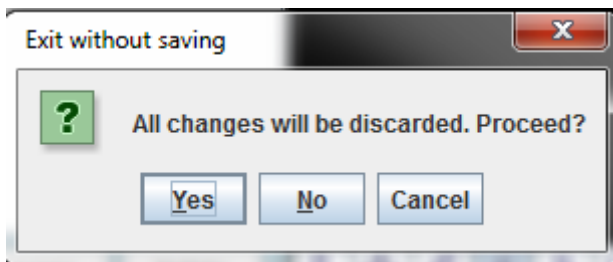
### 12.2.5 Έξοδος Από Την Εφαρμογή

Η τελευταία επιλογή που διατίθεται στον χρήστη, είναι βέβαια η έξοδος από την εφαρμογή. Για αυτό υπάρχει το κουμπί «Exit without saving» .



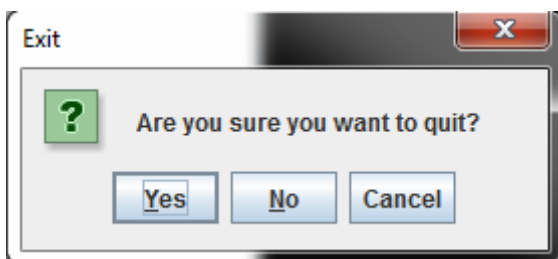
Εικόνα 80: Γραμμή εργαλείων- Επιλογή «Exit»

Με την ενεργοποίηση της εντολής η εφαρμογή ρωτάει τον χρήστη αν είναι βέβαιος για την απόφαση της εξόδου από την εφαρμογή, γιατί οι όποιες μη σωσμένες αλλαγές θα χαθούν. Συνεπώς, για να κρατήσει ότι έχει κάνει θα πρέπει πρώτα να έχει πατήσει την αποθήκευση και στην συνέχεια να πατήσει έξοδο.



Εικόνα 81: Επιβεβαίωση εξόδου με την επιλογή «Exit without Saving»

Πέρα από το συγκεκριμένο κουμπί στο toolbar, βέβαια, ο χρήστης μπορεί να κλείσει την εφαρμογή πατώντας απλά το «X» του παραθύρου, όπου και εκεί διερωτάται για επιβεβαίωση της απόφασης με σκοπό την αποφυγή των λαθών.



Εικόνα 82: Επιβεβαίωση εξόδου από το «X»

Αυτές είναι κάποιες από τις επιλογές που καλύπτονται από την γραμμή εργαλείων της εφαρμογής για την συγκεκριμένη διπλωματική διατριβή. Οι υπόλοιπες επιλογές, όπως άδειασμα της σκηνής, ανάρτηση ή επανεκτέλεση μιας ενέργεια, φόρτωση ενός υπάρχοντος εικονικού κόσμου, φόρτωση ενός τρισδιάστατου αντικειμένου στην σκηνή, πως βλέπω τον κώδικα ενός ver1 αρχείου και πως αποθηκεύω έναν εικονικό κόσμο, αφορούν το κομμάτι της σχεδίασης του εικονικού κόσμου και δεν αναλύονται περεταίρω σε αυτή την διπλωματική διατριβή.

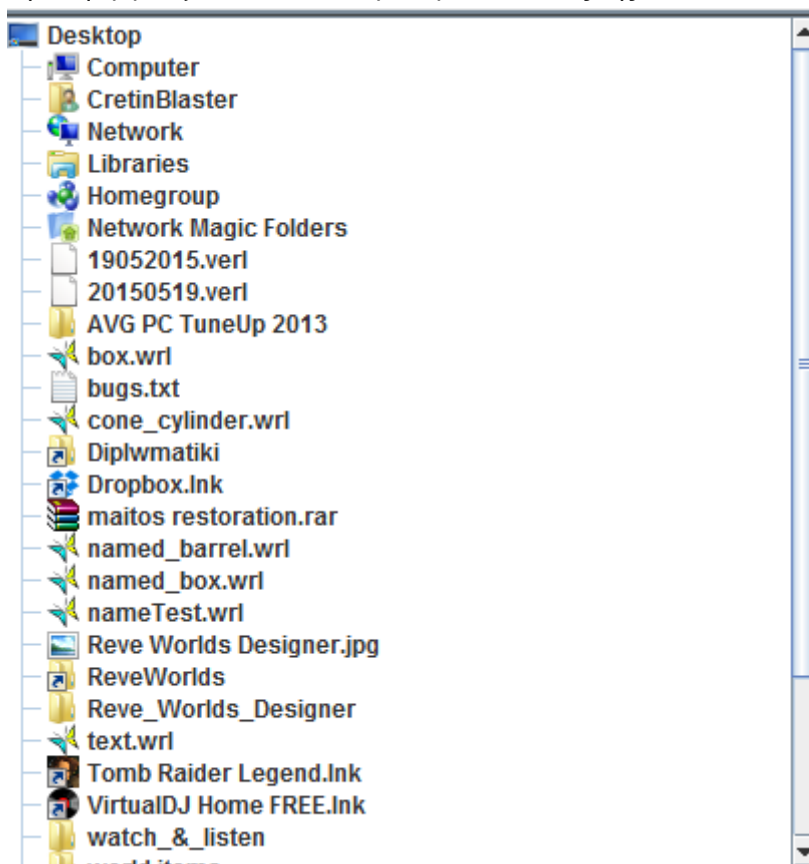
### 12.3 Κατάλογος Αρχείων Του Χρήστη

Όπως αναφέρθηκε και πιο πάνω έχει δοθεί η δυνατότητα στον χρήστη να μπορεί να ανοίξει ένα τρισδιάστατο αντικείμενο με δύο τρόπους.

Ο πρώτος τρόπος αναλύθηκε πιο πάνω και αφορούσε την εντολή «OpenFile» από την γραμμή εργαλείων.

Ο δεύτερος τρόπος αφορά ένα παραθυρικό περιβάλλον της εφαρμογής που έχει δημιουργηθεί και που απεικονίζονται σε αυτό ένας κατάλογος με τα αρχεία και τους φακέλους του χρήστη. Πιο αναλυτικά ο δεύτερος τρόπος για να μπορέσει ο χρήστης να ανοίξει ένα τρισδιάστατο αντικείμενο δίνεται μέσα από την εμφάνιση των φακέλων συστήματος του χρήστη, μέσω δηλαδή του FileSystemView της swing που εμφανίζεται σε ξεχωριστό παράθυρο στην εφαρμογή. Με αυτόν τον τρόπο, θεωρούμε πως είναι πιο άμεση και γρήγορη η πρόσβαση στα αρχεία του χρήστη. Ένα μεγάλο πλεονέκτημα που δημιουργήθηκε είναι ότι ο χρήστης έχοντας εύκολη πρόσβαση στα τρισδιάστατα αντικείμενα του δίνεται η δυνατότητα να μπορεί να εναλλάζει ανάμεσα σε αυτά και να τα παρατηρεί πιο γρήγορα και εύκολα.

Ο χρήστης μπορεί να εναλλάζει τα αντικείμενα χωρίς κάθε φορά να χρειάζεται να ανοίγει τους φακέλους από την γραμμή εργαλείων. Επιπλέον του επιτρέπεται παράλληλα η δυνατότητα να εκτελεί και άλλες ενέργειες, καθώς επιλέγοντας ένα αρχείο με το ποντίκι και πατώντας επάνω του τρεις φορές (γιατί 2 είναι οι default για το άνοιγμα ενός φακέλου) σε ένα μη συμβατό αρχείο, αυτό ανοίγει με το προεπιλεγμένο πρόγραμμα του λειτουργικού συστήματος, πχ επιλέγοντας 3 φορές ένα word αρχείο αυτό θα ανοίξει με το MSOfficeWord. Μόνο στην περίπτωση των verl αρχείων, η συμπεριφορά είναι αντίστοιχη με το OpenFile, δηλαδή εμφανίζεται σε άλλο παράθυρο ο κώδικας της verl.

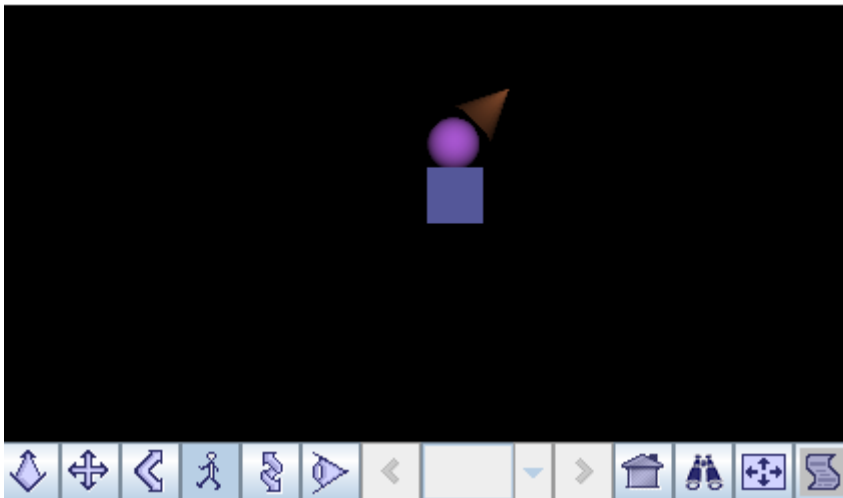


Εικόνα 83: Εμφάνιση φακέλων συστήματος για ευκολότερη περιήγηση του χρήστη στα αρχεία

## 12.4 Παράθυρο Απεικόνισης Τρισδιάστατων Αντικειμένων

Έχοντας αναλύσει την λειτουργία απεικόνισης των καταλόγων του χρήστη θα ξεκινήσουμε να παρουσιάζουμε ένα από τα βασικότερα παράθυρα της εφαρμογής. Το παράθυρο απεικόνισης τρισδιάστατων αντικειμένων. Πέρα από την επιλογή του αρχείου, είναι απαραίτητο ο χρήστης να βλέπει ποιο αντικείμενο ακριβώς απεικονίζεται στο εν λόγω αρχείο ώστε να αποφασίσει αν θέλει πράγματι να το προσθέσει στον κόσμο. Δίνεται η ευκολία στον χρήστη να πλοηγείται άμεσα στα αρχεία του συστήματος και να απεικονίζονται αυτά στο παράθυρο εμφάνισης τρισδιάστατων αντικειμένων.

Πιο αναλυτικά δημιουργήσαμε, λοιπόν, πάνω από το παράθυρο εμφάνισης φακέλων συστήματος, έναν `xj3dbrowser`, τον οποίο αποκαλούμε «ObjectViewport», όπου φορτώνεται το αντικείμενο που επιλέγεται κάθε φορά (Εικόνα 84). Επομένως, επιλέγοντας ένα κατάλληλο (δηλαδή `.wrl` ή `.x3dn`) αρχείο από το «OpenFile» ή επιλέγοντας μια φορά σε ένα τέτοιο αρχείο από το παράθυρο εμφάνισης φακέλων του συστήματος, το αντικείμενο φορτώνεται στον browser. Βέβαια, αυτό προϋποθέτει το αρχείο να είναι σωστά δομημένο ως `vml` ή `x3d` κώδικας και να μην είναι corrupt, διαφορετικά, όπως και στο Cortona 3DViewer η απεικόνισή του δεν καθίσταται δυνατή.



Εικόνα 84: ObjectViewport – Οθόνη απεικόνισης επιλεγμένου αντικειμένου

Τα ζητήματα που αντιμετωπίσαμε σε αυτό το σημείο ήταν διάφορα και μας δυσκόλεψαν αρκετά.

Πρώτα από όλα, για την απεικόνιση του αντικειμένου δεν αρκεί η δημιουργία ενός ExternalBrowser που παρέχει το βασικό πλαίσιο περιήγησης της σκηνής, αλλά απαιτείται και η δημιουργία της X3DScene σκηνής επί της οποίας τοποθετείται το αντικείμενο. Διαπιστώσαμε πως για να φορτώσουμε ένα άλλο αντικείμενο στον browser, δεν αρκούσε να αντικαταστήσουμε απλά τη σκηνή με το νέο αρχείο με τη χρήση της σχετικής μεθόδου `replaceWorld()` της `xj3d`, αλλά απαιτούσε να ξανά δημιουργήσουμε εκ νέου όλο τον browser και τη σκηνή, μαζί με το JPanel στο οποίο εμφανίζονται τα παράθυρα. Διαφορετικά, η μια σκηνή κρύβεται πίσω από την άλλη δημιουργώντας inconsistency τόσο στη λειτουργικότητα όσο και στο γραφικό περιβάλλον της διεπαφής. Επομένως, έπρεπε να ακολουθήσουμε διαφορετική προσέγγιση για την περίπτωση που επιλέγεται ένα αρχείο για πρώτη φορά και για την περίπτωση που έχει ήδη φορτωθεί ένα αρχείο και πάμε να φορτώσουμε άλλο.

Επιπρόσθετα, κατά την επιλογή ενός μη συμβατού αρχείου ή φακέλου, η σκηνή έπρεπε να καθαρίζεται, ώστε αυτό που βλέπει ο χρήστης να ανταποκρίνεται σε αυτό που επιλέγει κάθε φορά. Άρα έπρεπε εκ νέου να εκτελέσουμε την ίδια διαδικασία αφαίρεσης και εισαγωγής όλου του browser, καθώς η αντικατάσταση της σκηνής με `null` τιμή, παρότι αναφερόταν στο σχετικό tutorial, δεν δούλευε, ενώ η αφαίρεση των `rootnodes` του αντικειμένου από τη σκηνή – ώστε να

αποφευχθεί η διαδικασία προσθαφαίρεσης του browser - απαιτούσε διαφορετική προσέγγιση για την ενεργοποίηση ξανά της σκηνής, γεγονός που αύξανε την πολυπλοκότητα στον κώδικα.

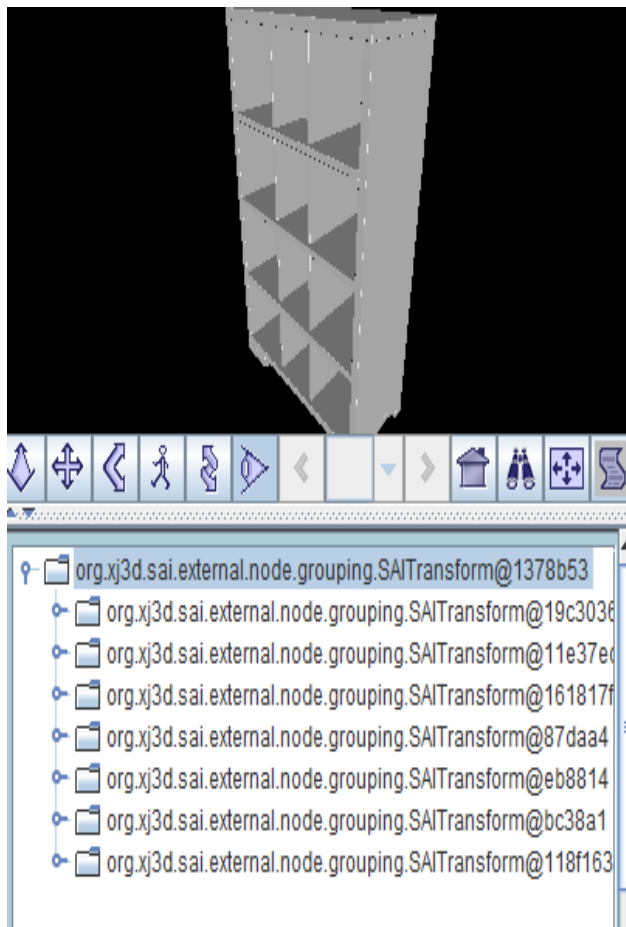
Τέλος, υπήρχε σοβαρό ζήτημα με τα verl αρχεία, καθώς ως άγνωστος τύπος και στο λειτουργικό σύστημα, η επιλογή τους αφαιρούσε ολοκληρωτικά το JPanel, μαζί με το toolbar που υπήρχε από πάνω του. Αδυνατώντας να εφαρμόσουμε την ίδια λύση για τα υπόλοιπα μη-συμβατά αρχεία, αποφασίσαμε στην περίπτωση που επιλέγεται verl αρχείο από τους φακέλους συστήματος, η εικόνα του browser να διατηρείται ίδια με την προηγούμενη, επομένως πρακτικά επαναφορτώνεται το αρχείο που υπήρχε πριν ή η σκηνή παραμένει κενή αν ένα μη-συμβατό αρχείο είχε επιλεγεί τελευταίο. Τα παραπάνω επιτεύχθηκαν με τη χρήση ενός counter που ελέγχει αν είναι η πρώτη φορά που φορτώνεται ένα αρχείο και αν το αρχείο αυτό είναι έγκυρο, καθώς και διατηρώντας σε μεταβλητές στοιχεία για το αρχείο που είχε φορτωθεί τελευταίο στον browser, για την περίπτωση που επιλεγεί κάποιο verl αρχείο.

## 12.5 Παράθυρο Γραφήματος Σκηνής

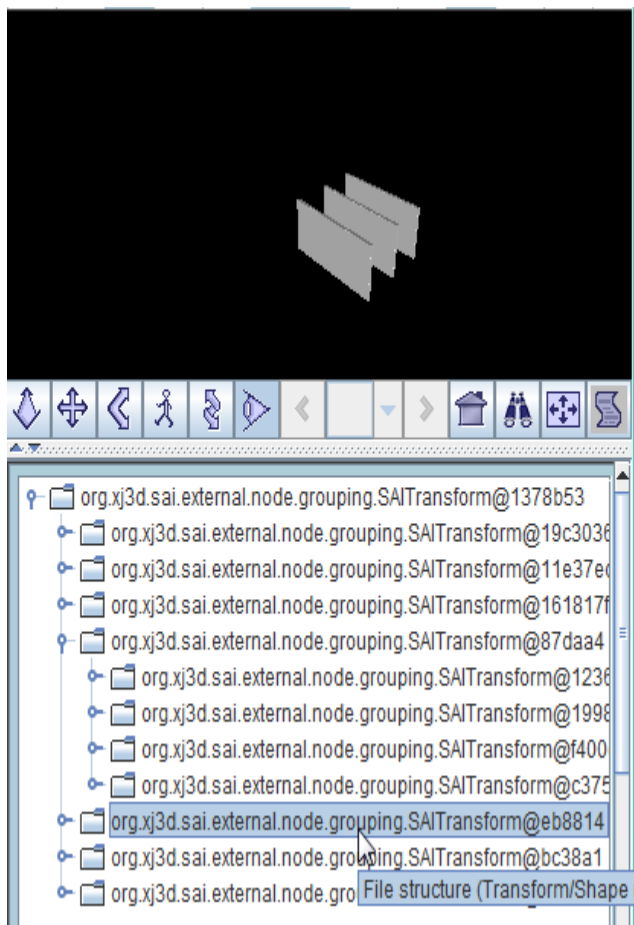
Στην συνέχεια με την επιλογή του τρισδιάστατου αντικειμένου και με την φόρτωση του στο παράθυρο απεικόνισης τρισδιάστατων αντικειμένων δημιουργείται και εμφανίζεται ένα επιπλέον παράθυρο, το παράθυρο γραφήματος σκηνής. Για εξοικονόμηση χώρου το συγκεκριμένο παράθυρο όταν δεν έχει επιλεγεί κάποιο αντικείμενο Vrm1 ή xj3d δεν εμφανίζεται. Με την επιλογή ενός αντικειμένου εμφανίζεται το παράθυρο και μέσα σε αυτό δημιουργείται το γράφημα σκηνής του.

Το γράφημα σκηνής είναι ένα από τα βασικότερα κομμάτια ολόκληρης της εφαρμογής. Έπρεπε να διασφαλίσουμε πως το αντικείμενο που επιλέγει ο χρήστης μπορεί να εισαχθεί στον κόσμο, δηλαδή πληρεί τις προδιαγραφές της verl και πρόκειται για έναν κόμβο DEFTtransform. Επομένως, έπρεπε να ελέγξουμε την εσωτερική δομή του αντικειμένου, δηλαδή τους X3Dκόμβους που το αποτελούν.

Παράλληλα, θέλαμε να δώσουμε στο χρήστη τη δυνατότητα να περιηγηθεί στο γράφημα σκηνής του αντικειμένου και να επιλέξει επιμέρους κόμβους. Μπορεί ο χρήστης να μην χρειαζότανε ολόκληρο το αντικείμενο αλλά ένα μέρος μόνο του αντικειμένου. Για παράδειγμα, αν το αντικείμενο ήταν μια βιβλιοθήκη (εικόνα 85), θέλαμε να δίνεται στον χρήστη η δυνατότητα να μπορεί να επιλέξει μόνο τα ράφια που επιθυμεί και που το αποτελούν για να τα προσθέσει στον κόσμο (Εικόνα 86).



Εικόνα 85: Εμφάνιση ολόκληρης της βιβλιοθήκης



**Εικόνα 86: Επιλογή μόνο των ραφιών**

Επομένως, κατά την επιλογή του αντικειμένου, ψάχνουμε τα rootnodes του, εκ των οποίων επιλέγουμε μόνο τα Transformnodes. Γίνεται έλεγχος αν τα Transformnodes έχουν children και άλλα Transformnodes, με αποτέλεσμα να επαναλάβουμε αναδρομικά την διαδικασία για κάθε Transformnode και children.

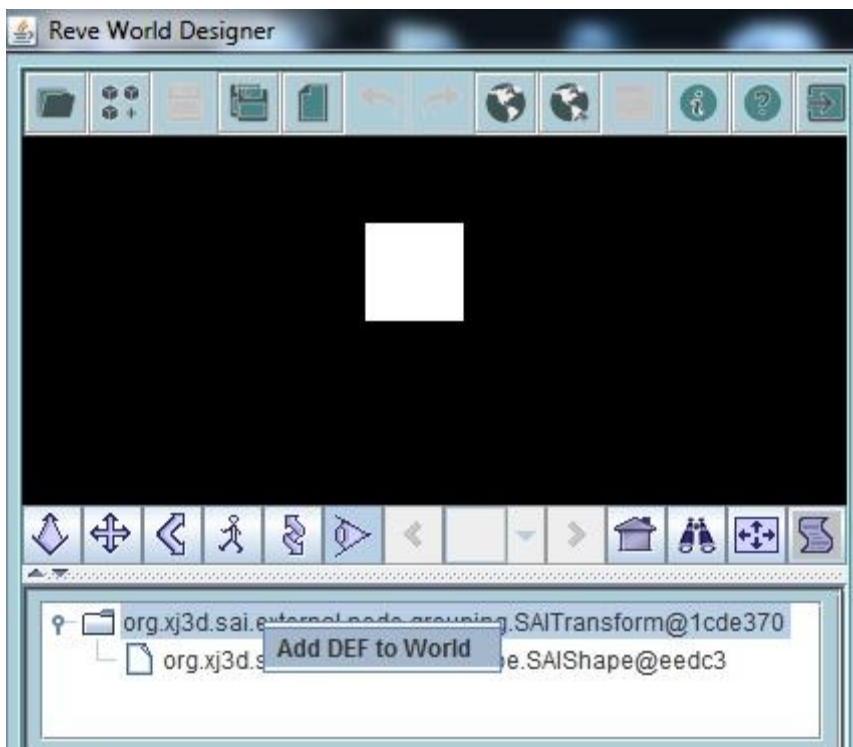
Δημιουργούμε μια αναδρομική συνάρτηση η οποία ελέγχει σε κάθε επανάληψη τον εαυτό της. Αυτό έχει ως αποτέλεσμα να εμφανίζουμε ένα γράφημα σκηνης του αντικειμένου που λαμβάνει υπόψη την εσωτερική του δομή για Transform και Shape nodes. Κατά την επιλογή των εσωτερικών κόμβων του γραφήματος σκηνης, ο browser αδειάζει από το rootnodes που έχει με την μέθοδο removeRootNode() της xj3d και ξαναγεμίζει με τον επιλεγμένο κόμβο, καθιστώντας αυτόν ως rootnode με τη μέθοδο addRootNode() της xj3d.

Σε αυτό το σημείο, οφείλουμε να πούμε πως έπρεπε να τοποθετήσουμε το γράφημα σκηνης του αντικειμένου σε ένα σημείο χωρικά κοντά με την αναπαράστασή του, χωρίς ωστόσο να καταλαμβάνει ένα δεδομένο μέγεθος κάθε φορά στην οθόνη, δεδομένου ότι ο χώρος ήταν περιορισμένος και δε θέλαμε να τον υπερφορτώσουμε με πληροφορία. Επομένως, αποφασίσαμε να προσθέσουμε το γράφημα σκηνης σε ένα JPanel το οποίο εμφανίζεται μόνο κατά την επιλογή ενός αντικειμένου και καταλαμβάνει ένα χώρο σχετικό με το μέγεθος του γραφήματος σκηνης του. Αυτό, βέβαια, είχε ως αποτέλεσμα να πρέπει να καθαρίζουμε το γράφημα σκηνης κατά την επιλογή διαφορετικού αντικειμένου ή να το αφαιρούμε τελείως κατά την επιλογή μη-συμβατού αρχείου.

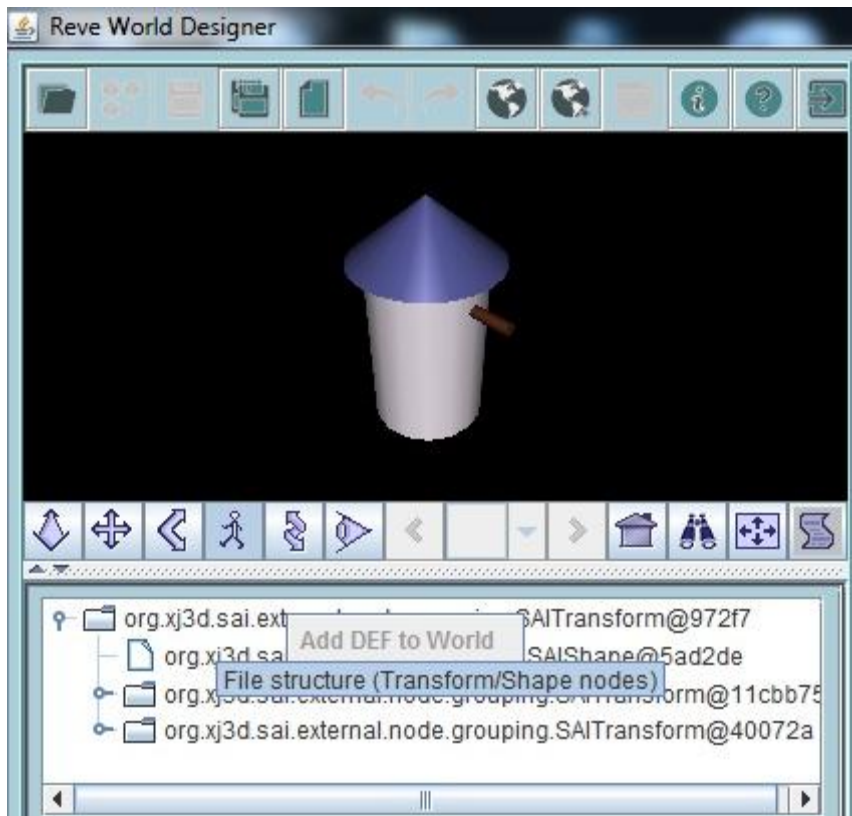
Για την εισαγωγή του επιλεγμένου κόμβου στην σκηνή του κόσμου αναφέραμε πιο πάνω την επιλογή που έχει δοθεί μέσω του κεντρικού μενού. Εκτός όμως από την συγκεκριμένη επιλογή δίνεται στον χρήστη η επιλογή της εισαγωγής μέσω του δεξιού κλικ, επιλέγοντας τον



κόμβο που θέλουμε να εισάγουμε με την ονομασία “AddDEFtoWorld”. **Add DEF to World**. Και στις δυο περιπτώσεις εισαγωγής ενός κόμβου, ελέγχεται αν ο κόμβος που επιλέγει ο χρήστης είναι DEFTransform. Αν και μόνο αν έχει επιλεγεί ένας DEFTransform κόμβος (Εικόνα 87) και μόνο σε αυτή την περίπτωση ενεργοποιείται η δυνατότητα προσθήκης του. Αλλιώς η επιλογή «AddDEFtoWorld» παραμένει γκριζαρισμένη και μη προσβάσιμη (Εικόνα 88).



Εικόνα 87: Αντικείμενο με DEFTransform κόμβο



Εικόνα 88: Αντικείμενο χωρίς DEFTransform κόμβο. Οι επιλογές προσθήκης είναι γκριζαρισμένες για Shape κόμβους.

## 12.6 Ορισμός Του Item

Το επόμενο ζητούμενο που έπρεπε να αντιμετωπίσουμε είναι να εισάγουμε τους DEF X3DNode κόμβους στον εικονικό κόσμο. Εδώ είναι που γίνεται και η σύνδεση μεταξύ των δυο μεταπτυχιακών διατριβών, μέσω της μετατροπής ενός κόμβου σε item για να προχωρήσουμε στη σχεδίαση του εικονικού κόσμου. Στην παρακάτω εικόνα φαίνεται η προσθήκη των επιλεγμένων αντικειμένων στην λίστα «World Items» (Εικόνα 89). Στην συγκεκριμένη λίστα εισάγονται τα αντικείμενα ή μέρος των αντικειμένων που έχει επιλέξει ο χρήστης. Με την εισαγωγή ενός κεντρικού παραθύρου απεικόνισης τρισδιάστατων αντικειμένων αυτά τα αντικείμενα εμφανίζονται στον εικονικό κόσμο μας.



**Εικόνα 89: Λίστα «World Items»**

Για αυτό το σκοπό, δημιουργούμε μια ξεχωριστή κλάση (Item.java), όπου α) κρατάμε τις πληροφορίες από τον X3DNodeκόμβο που χαρακτηρίζουν ένα item, όπως είναι το όνομα του DEF κόμβου (που μας χρειάζεται για το virtual source) και το όνομα και η τοποθεσία του αρχείου από όπου παίρνουμε το αντικείμενο(που μας χρειάζεται για το itemGroup source) και β) δημιουργούμε μεθόδους για να ορίσουμε το όνομα ενός item καθώς και τα βασικά του συστατικά (virtual model, access model, semantic model).

Πέρα από τον κόμβο για να δημιουργήσουμε ένα item, χρειαζόμαστε και μια σκηνή, που θα είναι και η σκηνή του κόσμου μας. Αυτό, γιατί το item προϋποθέτει έναν εξωτερικό Transform κόμβο, που θα είναι ο κόμβος-ρίζα του γραφήματος σκηνής που αποτελεί το αντικείμενο, πάνω

στον οποίον θα γίνονται όλα τα transformations. Χρειαζόμαστε, δηλαδή, την X3DScene σκηνή, πέρα από την απεικόνιση του κόσμου, για να χρησιμοποιήσουμε την μέθοδο createNode("Transform") του xj3dπου φτιάχνει έναν Transform κόμβο, ώστε ως παιδί του να ορίσουμε τον X3Dκόμβο που επιλέγεται από το γράφημα σκηνής. Έπρεπε, λοιπόν, να αρχικοποιήσουμε τη σκηνή και για αυτό τον σκοπό χρησιμοποιούμε ένα dummy vrml αρχείο, το test. wrl, φορτώνοντάς το για να δημιουργήσουμε τη σκηνή και αφαιρώντας αμέσως μετά το περιεχόμενό του, ώστε η σκηνή να μείνει κενή για χρήση. Παρακάτω ο κώδικας για το βασικό constructor του item:

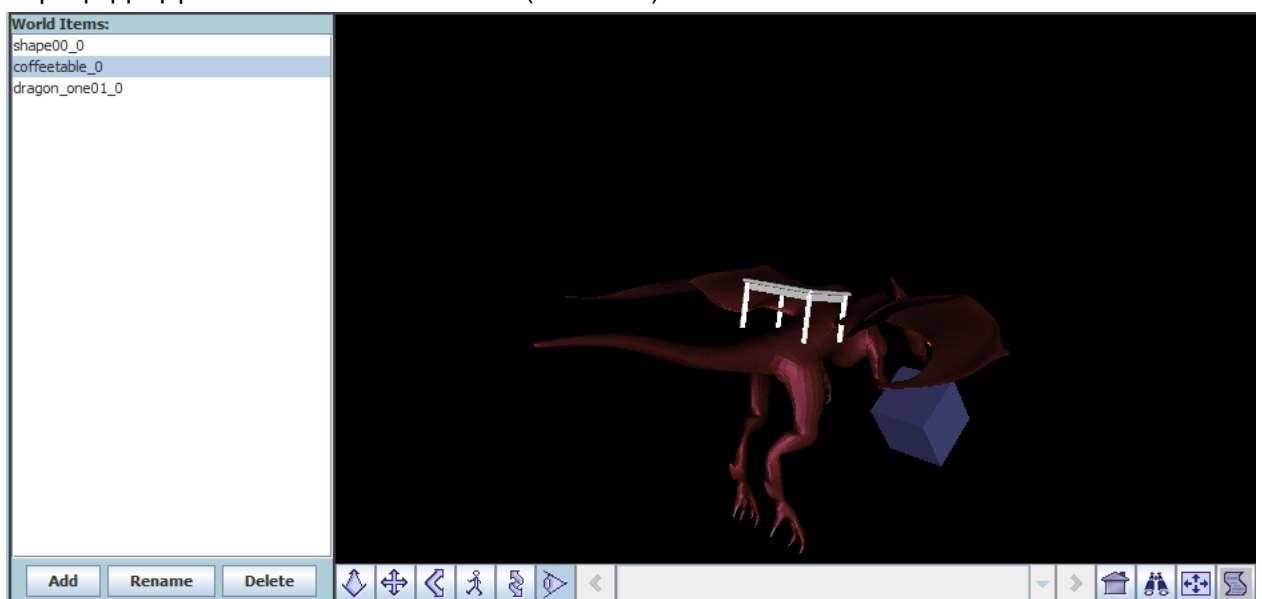
```
//Item constructor with def name
public Item(X3DScene mainScene, X3DNode node, String name, String path){
    this.mainScene = mainScene;
    this.copy = node;
    this.location = path;
    //////////////////////////////////////
    SFVec3f translation = (SFVec3f)node.getField("translation");
    System.out.println("Original translation "+translation.toString());
    //////////////////////////////////////
    //Create a new root Transform node to add the object under it
    //in order to be able to handle its transformations separately
    transform = mainScene.createNode("Transform");
    children = (MFNode)(transform.getField("children"));
    children.append(node);
    //Create the item and save its DEF name
    createItem(transform,name);
    //Give a default name to the item added
    setDefaultName();
    //Set a default semantic and access model with null values
    setDefaultSemanticModel();
    setDefaultAccessModel();
}
//end constructor
```

Έχοντας, λοιπόν, 1) τον επιλεγμένο DEF κόμβο, 2) τη σκηνή του κόσμου και 3) τον ορισμό του item, μπορέσαμε να προσθέσουμε ένα αντικείμενο ως item στην σκηνή, εμπλουτίζοντας τον κώδικα των σχετικών κουμπιών (AddNode/DEFtoWorld).

Σε αυτό το σημείο, οφείλουμε να πούμε πως υπάρχει ένα ουσιαστικό πρόβλημα που δεν λύθηκε στα πλαίσια αυτής της διπλωματικής διατριβής. Η εισαγωγή ενός item στην σκηνή δεν το ευθυγραμμίζει, όπως η Reve, στην αρχή των αξόνων του κόσμου, δηλαδή δεν τοποθετείται στο σημείο (0,0,0) στην οθόνη του κόσμου αλλά το αντικείμενο διατηρεί το transformation που έχει από τον VRML κώδικα, με αποτέλεσμα να εμφανίζεται σε εκείνες τις συντεταγμένες στον κόσμο. Αυτό σημαίνει πως το εξωτερικό Transform που προστίθεται για την λειτουργία του virtual model, λαμβάνει ως σημείο αναφοράς το σημείο που βρίσκεται κάθε φορά το εισαγόμενο αντικείμενο και όχι την αρχή των αξόνων. Επομένως, το virtual model είναι σχετικό με την θέση και το συνολικό transformation κάθε αντικειμένου και όχι με την αρχή των αξόνων. Παρότι έγινε προσπάθεια το αντικείμενο να ευθυγραμμιστεί στην αρχή των αξόνων, μέσω προσθαφαίρεσης των translations και των rotation του, αυτό δεν κατέστη εφικτό καθώς δημιουργούσε προβλήματα, με το αντικείμενο να έχει τυχαία συμπεριφορά ως προς την θέση του στον κόσμο. Πέραν τούτου, η συγκεκριμένη προσθαφαίρεση θα έπρεπε να γίνει σε όλα τα εσωτερικά transformations από τα οποία αποτελείται ένα αντικείμενο, ενώ αν υπήρχε και scale

στο αντικείμενο δημιουργούσε περαιτέρω προβλήματα. Δεδομένου ότι ο κ. Αναστασάκης έχει υλοποιημένο κώδικα για να επιτευχθεί η ευθυγράμμιση στην αρχή των αξόνων, αποφασίστηκε αυτή η διόρθωση να γίνει σε δεύτερο χρόνο είτε από τον ίδιο είτε από άλλον συνάδελφο, εφόσον απαιτούσε αρκετό χρόνο και η διπλωματική διατριβή προσφέρεται για περαιτέρω συνέχεια. Ωστόσο, υπάρχουν σχόλια στον κώδικα όπου θεωρούμε πως πρέπει να τοποθετηθεί το συγκεκριμένο κομμάτι κώδικα, συγκεκριμένα στον constructor του Itemclass.

Επιστρέφοντας στο προηγούμενο θέμα, κατά την εισαγωγή του αντικειμένου στην σκηνή του κόσμου ενημερώνουμε δυο αλληλένδετες δομές δεδομένων όπου πρέπει να αποθηκεύουμε την πληροφορία του κόσμου, που είναι δυο λίστες, μια με τα ονόματα των αντικειμένων και μια με τα items που υπάρχουν στον κόσμο. Αυτές οι δομές δεδομένων μοναδικοποιούνται μέσω του ονόματος του αντικειμένου που προσδιορίζει την ταυτότητά του. Η διάκριση μεταξύ των δυο δομών έγινε καθαρά για να μπορεί ο χρήστης να επιλέξει ένα αντικείμενο βάσει του ονόματός του και να το χειριστεί από τη σχετική JList που εμφανίζεται στην εφαρμογή υπό το JLabel "WorldItems" (Εικόνα 90).



Εικόνα 90: Τα ονόματα των items (αριστερά) και τα items στην σκηνή (δεξιά)

## Κεφάλαιο 13 - Συμπεράσματα

Η παρούσα διπλωματική διατριβή είχε ως στόχο την ανάπτυξη μιας εφαρμογής μέσω της οποίας θα μπορούσε ο χρήστης με την επιλογή των κατάλληλων τρισδιάστατων αντικειμένων να ξεκινήσει την σύνθεση ενός εικονικού κόσμου. Για να μπορέσει όμως να ολοκληρωθεί η σύνθεση του εικονικού κόσμου έπρεπε το κομμάτι της συγκεκριμένης διπλωματικής διατριβής να συνδυαστεί και να ολοκληρωθεί με την διπλωματική διατριβή της συναδέρφου Έντα Τσελεμέγκου που είχε ως περιεχόμενο την σχεδίαση εικονικών κόσμων αξιοποιώντας την αναπαράσταση Reve. Ο στόχος εξ αρχής ήταν η εφαρμογή να μπορεί να αποτελέσει την βάση για μια επεκτάσιμη πλατφόρμα σύνθεσης εικονικών κόσμων με αντικείμενα vml και x3d.

Η εφαρμογή πέτυχε τον στόχο αυτό, χωρίς όμως να είναι βέλτιστη ούτε από άποψη πολυπλοκότητας κώδικα ούτε από άποψη κάλυψης πλήρους λειτουργικότητας της αναπαράστασης Reve. Για αυτό τον λόγο, άλλωστε, η υλοποίηση εστίασε στην φυσική αναπαράσταση ενός κόσμου και όχι τόσο στην σημασιολογική αναπαράσταση του και στην δυνατότητα αλληλεπίδρασης με αυτόν, παρότι έγινε μια πρώτη προσέγγιση για την ανάπτυξη των εν λόγω μοντέλων.

Εξετάζοντας την ανάλυση απαιτήσεων και παρατηρώντας το τι θέλαμε ιδανικά να καλύψουμε και να υλοποιήσουμε στα πλαίσια αυτής της εφαρμογής, μπορούμε να πούμε πως κατά την σχεδίαση ένα μεγάλο κομμάτι υλοποιήθηκε πλήρως και κατά την λειτουργικότητα ένα μεγάλο κομμάτι επιτεύχθηκε, και αφορά το κομμάτι που αποτελεί και την βασική λειτουργικότητα της τρέχουσας διατριβής, δηλαδή την σχεδίαση ενός εικονικού κόσμου. Σίγουρα υπάρχουν λειτουργίες που θα μπορούσαν να προστεθούν, να βελτιώσουν την εφαρμογή και να την κάνουν πληρέστερη.

Είναι εμφανές ότι η εφαρμογή διαθέτει ένα φιλικό προς τον χρήστη περιβάλλον λειτουργίας, με την ανάπτυξη ενός friendly user interface που περιλαμβάνει ένα σύνολο κουμπιών για τις εντολές που μπορούν να εκτελεστούν. Τα κουμπιά αυτά διαθέτουν περιγραφή για το τι κάνουν ή οδηγίες για την ενεργοποίησή τους από το πληκτρολόγιο και τοποθετούνται σε θέση σχετική με την λειτουργικότητα που επιτελούν, ώστε να είναι εύχρηστα και κατανοητά από τον χρήστη. Έτσι, έχουμε κουμπιά στο toolbar για τις κύριες λειτουργίες της εφαρμογής, στην λίστα των αντικειμένων που υπάρχουν στην σκηνή του κόσμου για τις βασικές εντολές που μπορούν να εκτελεστούν πάνω σε ένα item (add, rename, delete), σε καθένα από τα μοντέλα αναπαράστασης ενός item, καθώς και σε πτυσσόμενα μενού (που εμφανίζεται με δεξί κλικ πάνω στο item) για πιο γρήγορη εκτέλεση των εντολών και για να δώσουμε στον χρήστη μια πληθώρα επιλογών χωρίς να υπερφορτώσουμε το παράθυρο με πολλή πληροφορία.

Αυτό που πραγματικά πετύχαμε είναι να μπορεί να γίνεται εισαγωγή ενός τρισδιάστατου αντικειμένου στον εικονικό κόσμο της εφαρμογής, και στην συνέχεια να δίνεται η ευκαιρία στον χρήστη να το επεξεργαστεί και να το φέρει στα μέτρα του. Πιο λεπτομερώς πετύχαμε αρχικά να δείχνουμε ποιο item είναι αυτό που επιλέγεται από την λίστα με τα αντικείμενα της σκηνής, αξιοποιώντας την scale ιδιότητά του. Ακόμα, επιτύχαμε να δίνεται η ευκαιρία στον χρήστη να διαγράψει ένα item από την σκηνή, να δημιουργεί αντίγραφα κάποιου αντικειμένου και να το μετονομάζει, διατηρώντας όμως πάντα την ταυτότητά του, μέσα από μοναδικοποίηση του ονόματός του. Επιπλέον για να μπορεί ο χρήστης να έχει μια πλήρη εικόνα για τον κόμβο-ρίζα του item η εφαρμογή δίνει την επιλογή στον χρήστη να βλέπει το επιλεγμένο αντικείμενο να περιστρέφεται γύρω από τον εαυτό του. Εκτός όμως από αυτές τις βασικές επιλογές προσθέσαμε την δυνατότητα ο χρήστης να ορίζει ως 'real' ή 'unreal' ένα item στην σκηνή, με τα σημασιολογικά αποτελέσματα που αυτό έχει στην Reve Worlds, ενώ εμφανίζουμε σε ξεχωριστό παράθυρο τα βασικά στοιχεία που αποτελούν ένα item, ώστε ο χρήστης να έχει μια καλύτερη εικόνα για την προέλευσή του και τις ιδιότητές του. Πέραν τούτου, εισάγαμε την δυνατότητα ο χρήστης να μπορεί να αναιρεί μια ενέργεια, σε περίπτωση λάθους, καθώς και να επανεκτελεί την ενέργεια που μόλις αναίρεσε. Τέλος, ο χρήστης μπορεί να καθαρίζει την σκηνή ξεκινώντας μια νέα σχεδίαση από την αρχή ή να προσθέτει έναν εικονικό κόσμο στον υπάρχον που σχεδιάζει, για να τον εμπλουτίσει πιο γρήγορα, συγχωνεύοντας πλήθος αντικειμένων σε μια νέα σχεδίαση.

Θέλοντας να δημιουργήσουμε μια φιλική προς τον χρήστη εφαρμογή σκοπός ήταν η δομή της εφαρμογής να διευκολύνει τον χρήστη στην επιλογή αντικειμένων που θα εισάγει στον

κόσμο. Αυτό επιτεύχθηκε με το να αποτελείται η εφαρμογή από επιμέρους παράθυρα με ξεχωριστή λειτουργικότητα το κάθε ένα. Η εφαρμογή από την μια πλευρά εμφανίζει τα αντικείμενα που επιλέγει ο χρήστης και την εσωτερική τους δομή, επιτρέποντάς τον να τα επεξεργαστεί συνολικά και ελέγχοντας ποιους κόμβους μπορεί και θέλει να εισάγει στον κόσμο βάσει των προδιαγραφών της *vef* κι από την άλλη πλευρά, διατηρείται μια αυτονομία, εφόσον ο χρήστης μπορεί να ασχοληθεί ανεξάρτητα με την επεξεργασία ενός εικονικού κόσμου.

Επίσης, καταφέραμε να διατηρήσουμε την αυτονομία των διαφορετικών διαστάσεων που χαρακτηρίζουν ένα *item* μέσα από την αναπαράσταση του *virtual*, *semantic* και *access model* του σε διακριτές θέσεις στην παραθυρική εφαρμογή και τη δυνατότητα παρέμβασης του χρήστη σε αυτά. Ως εκ τούτου, επιτύχαμε τον βασικό στόχο να σχεδιάσουμε έναν εικονικό κόσμο από ετερόκλητα αντικείμενα ή κομμάτια αντικειμένων από το γράφημα σκηνής τους, αλλάζοντας την θέση τους στο χώρο, το μέγεθός τους ή και την περιστροφή τους, δηλαδή μεταβάλλοντας το *virtual model* τους. Ακόμα, επιτρέψαμε στον χρήστη να ορίσει ένα υποτυπώδες *semantic* ή *access model* σε ένα *item*, για να βεβαιωθούμε ότι και αυτή η λειτουργικότητα είναι δυνατή μέσα από την τρέχουσα εφαρμογή, χωρίς να επεκταθούμε περισσότερο σε αυτή την έκφανση του *item*.

Τέλος έχει πολύ μεγάλο ενδιαφέρον το γεγονός ότι μέσα από την εφαρμογή μπορεί να γίνει η εξαγωγή ενός εικονικού κόσμου με την αναπαράσταση *Reve*, δημιουργώντας ένα μοναδικό *vef* αρχείο που αποθηκεύεται στην τοποθεσία που ορίζει ο χρήστης. Με αυτή την λειτουργικότητα δύναται η δυνατότητα στον χρήστη να μπορεί να βλέπει απευθείας τον *vef* κώδικα που αναπαριστά την σκηνή του. Με την ανάποδη λογική δόθηκε η δυνατότητα στον χρήστη να φορτώσει, αντιστρόφως, έναν εικονικό κόσμο που δημιουργείται από την εφαρμογή, εισάγοντας ένα *vef* αρχείο σε αυτήν.

Όμως όπως είναι λογικό, κάποιες ιδέες που αναφέραμε στην ανάλυση απαιτήσεων δεν καταφέραμε να τις πραγματοποιήσουμε, είτε επειδή το μέγεθος της δυσκολίας τους δεν κατέστησε εφικτή την υλοποίησή τους στην παρούσα φάση, είτε επειδή θεωρήθηκαν δευτερεύοντα ως προς τον πρωτεύον στόχο είτε γιατί θα παγιδευόμασταν σε έναν φαύλο κύκλο διαρκούς βελτίωσης της εφαρμογής, την στιγμή που έπρεπε, για αντικειμενικούς λόγους, να μπουν συγκεκριμένα χρονικά πλαίσια για την ολοκλήρωσή της. Παρακάτω θα αναφέρουμε κάποια βασικά πράγματα που δεν καταφέραμε να πραγματοποιήσουμε βάσει της ανάλυσης απαιτήσεων ή που δεν αναφέραμε στην ανάλυση απαιτήσεων, αλλά στην πορεία θεωρούμε πως θα ήταν ενδιαφέρον και εφικτό να πραγματοποιηθούν για την βελτιστοποίηση, τον εμπλουτισμό και την επέκταση της εφαρμογής.

Το πρώτο και κύριο που επιβάλλεται να διορθωθεί για την σωστή αναπαράσταση του κόσμου στην πλατφόρμα *Reve Worlds* και αποτελεί έλλειψη της τρέχουσας υλοποίησης είναι να εισαχθεί ο κώδικας του *k*. Ανασασάκη στην εφαρμογή, κατά την δημιουργία ενός *item*, ώστε το αντικείμενο που προστίθεται στον κόσμο να εμφανίζεται στην αρχή των αξόνων του κόσμου, έχοντας αφαιρέσει το εσωτερικό του *translation* και *rotation*. Είναι μέγιστης σημασίας να ευθυγραμμιστεί το αντικείμενο με την αρχή των αξόνων του κόσμου και το *virtual model* του να έχει ως σύστημα αναφοράς το σύστημα συντεταγμένων του κόσμου και όχι το τοπικό σύστημα συντεταγμένων του αντικειμένου. Επίσης, συνδυαστικά με αυτό, κατά την αποθήκευση του κόσμου, η παράμετρος “*fit*” του *item* πρέπει να πάρει την τιμή “*true*” και όχι “*false*”, για να εφαρμόζεται το παραπάνω και στο *Reve Worlds* κατά την φόρτωση του κόσμου.

Δεύτερον, πρέπει να προστεθεί η δυνατότητα πέρα από *DEF Transform* κόμβους να εισάγονται και *Shape* κόμβοι στην σκηνή. Ακόμα θα ήταν ιδιαίτερα ενδιαφέρον να δίνεται η δυνατότητα στον χρήστη για έναν επιπλέον τρόπο εισαγωγής αντικειμένων στον εικονικό κόσμο. Ο επιπλέον τρόπος που θα μπορούσε να εισαχθεί είναι να κάνει *drag and drop* ένα κόμβο στην σκηνή, αξιοποιώντας την σχετική δυνατότητα της *swing* και εμφανίζοντας σχετικό μήνυμα όταν αυτός ο κόμβος δεν πληρεί τις προϋποθέσεις προσθήκης στον εικονικό κόσμο.

Τρίτον, θα μπορούσε να βελτιωθεί το υπάρχον *user interface*, προσαρμόζοντας καλύτερα το παράθυρο σε οθόνες μικρότερης ή μεγαλύτερης ανάλυσης ή δίνοντας την δυνατότητα στον χρήστη να αυξομειώσει το μέγεθος των διακριτών παραθυρικών τμημάτων της εφαρμογής, όπως του κόσμου ή του *Object Navigator*.

Τέταρτον, μια ενδιαφέρουσα και πολύ χρήσιμη βελτίωση θα ήταν να αυξηθεί το πλήθος των undo-redo ενεργειών που επιτρέπονται στον χρήστη, αυξάνοντας το limit του undoManager σε έναν λογικό αριθμό. Ο αριθμός αυτός θα πρέπει να μελετηθεί σε σχέση με άλλα εργαλεία σχεδίασης εικονικών κόσμων αλλά και τις απαιτήσεις σε μνήμη που μπορεί να υπάρχουν για αυτή την περίπτωση από την εφαρμογή.

Πέμπτον, μπορεί να δοθεί η δυνατότητα επεξεργασίας πολλαπλών αντικειμένων ταυτόχρονα, όπως είναι η περίπτωση επιλογής πολλών items από την λίστα της σκηνής και την δημιουργία αντιγράφων τους με το πάτημα του 'Add' μια φορά ή την ταυτόχρονη διαγραφή τους με το πάτημα του 'Delete' μια φορά. Σε αντίθεση με την τρέχουσα υλοποίηση, που κρατάει σε μια Item μεταβλητή το επιλεγμένο item, εδώ θα πρέπει να διατηρούνται τα επιλεγμένα αντικείμενα σε μια λίστα ή σε μια στοίβα από Items για να μπορούν να επεξεργαστούν ταυτόχρονα. Επίσης, θα πρέπει αυτή η δυνατότητα να μπορεί να αναιρεθεί ή να επανεκτελεστεί μέσω του undo-redo μηχανισμού, γιατί διαφορετικά μπορεί να έχει τραγικές συνέπειες, εξού και η σημασία της προσωρινής αποθήκευσής τους σε μια ευέλικτη δομή δεδομένων.

Έκτον, η επίτευξη της παραπάνω δυνατότητας θα ανοίξει τον δρόμο και για την αναίρεση της προσθήκης ενός κόσμου (Add World) στον υπάρχον, γλιτώνοντας τον χρήστη από χρονοβόρα διαδικασία της διαγραφής ένα προς ένα τα αντικείμενα που προστέθηκαν με τον νέο κόσμο αν στην πορεία μετανιώσει αυτή την απόφαση ή αν κατά λάθος προσθέσει άλλον κόσμο από τον επιθυμητό. Σε σχέση με αυτό, θα μπορούσε να δίνεται η δυνατότητα να εμφανίζεται ο κόσμος που ο χρήστης θέλει να προσθέσει στον υπάρχον σε ένα νέο παράθυρο, όπως συμβαίνει με την εμφάνιση του περιστρεφόμενου αντικειμένου σε ένα νέο παράθυρο.

Έβδομον, όσον αφορά το semantic και access model, μπορεί και θα εξυπηρετούσε να προστεθεί έλεγχος για την περίπτωση λογικών σφαλμάτων στον ορισμό παραμέτρων, ώστε να αποφεύγονται λάθη κατά την φόρτωση στην πλατφόρμα Reve Worlds. Τέτοια λογικά σφάλματα είναι η εισαγωγή κόμβων που δεν ορίζονται στην δομή του αντικειμένου ή η εισαγωγή κόμβων που ορίζονται αλλά δεν συνδυάζονται με την function που πάει να αποθηκεύσει ο χρήστης. Επίσης, θα πρέπει να προστεθεί κάποιο κουμπί στο semantic και access model, όπως ένα σύμβολο '+', που θα επιτρέπει στον χρήστη να προσθέσει παραπάνω από ένα symbol ή access point στο επιλεγμένο item.

Όγδων, θα μπορούσε να αλλάξει το φόντος της σκηνής από μαύρος σε λευκός και να προστεθούν grids κατά μήκος των (x,y,z) αξόνων ώστε ο χρήστης να αντιλαμβάνεται καλύτερα πώς πρέπει να μετατοπίσει ένα αντικείμενο για να επιτύχει το επιθυμητό αποτέλεσμα. Συνδυαστικά με αυτό, θα μπορούσαν να προστεθούν κάποια διαφορετικά default viewpoints στην σκηνή του κόσμου, ώστε ο χρήστης να μπορεί να αλλάξει εύκολα και γρήγορα οπτική γωνία στον κόσμο, όπως συμβαίνει στην πλατφόρμα Reve Worlds.

Ένατον, θα πρέπει να εμπλουτιστεί ο exception μηχανισμός της φόρτωσης ενός κόσμου, ώστε να μπορεί πιο εύκολα να φορτωθεί ένα verl αρχείο που έχει γραφτεί με το χέρι και δεν έχει παραχθεί από την εφαρμογή, για να μην υπάρχουν σφάλματα, αλλά και για να αυξηθεί το εύρος των εικονικών κόσμων που πράγματι μπορούν να φορτωθούν χωρίς να περιορίζεται σε αυτούς που έχει φτιάξει η εφαρμογή. Αυτό θα ανοίξει την δυνατότητα ο χρήστης να μπορεί να επεξεργάζεται και χειροκίνητα τον κώδικα που παράγεται κατά την αποθήκευση ενός κόσμου, παρότι αυτό με την σειρά του συνεπάγεται την εκ νέου φόρτωση του κόσμου με τις νέες αλλαγές.

Τέλος, θα μπορούσε να δίνεται η δυνατότητα ο κόσμος να γίνεται preview μετά την αποθήκευσή του απευθείας στην πλατφόρμα Reve Worlds, ενώ μια άλλη ιδέα που μας άρεσε, αν και αναγνωρίζουμε πως η υλοποίησή της είναι αρκετά δύσκολη, είναι η εφαρμογή να επιτρέπει την σύνδεση πολλαπλών χρηστών σε αυτήν που θα σχεδιάζουν παράλληλα έναν εικονικό κόσμο. Αυτό θα είχε ιδιαίτερο ενδιαφέρον για τη χρήση της σε ένα εργαστηριακό μάθημα όπου πολλοί φοιτητές καλούνται να συνεργαστούν για αυτόν τον σκοπό.

Όπως αναλύθηκε στα προηγούμενα κεφάλαια και γίνεται κατανοητό από τα παραπάνω, συναντήσαμε αρκετές δυσκολίες κατά την υλοποίηση της εφαρμογής, οι οποίες δεν ήταν αμελητέες στον χρόνο που χρειαστήκαμε για να ολοκληρώσουμε την παρούσα διατριβή και μάλιστα κάποιες από αυτές δεν καταφέραμε να τις επιλύσουμε. Άλλωστε, το γεγονός ότι η εφαρμογή «έσπασε» σε δυο κομμάτια, όπως και οι ελλείψεις που αναδείχθηκαν στα πλαίσια της



υλοποίησης, μαρτυρούν το μέγεθος της δουλειάς που κρυβόταν από πίσω καθώς και το σύνολο των δυσκολιών με τις οποίες ήρθαμε αντιμέτωποι.

Παρότι, λοιπόν, θεωρητικά ο χρήστης μπορεί με μια πρώτη ματιά να μην εντυπωσιαστεί με το πλήθος των δυνατοτήτων που δίνονται από το Reve World Designer, οφείλει να έχει υπόψη του πως η υλοποίηση έπρεπε να λάβει και να προβλέψει ένα σύνολο παραγόντων και να συνδυάσει ένα σύνολο διαφορετικών εντολών που πολλές φορές οδηγούσαν την εφαρμογή σε απρόβλεπτα σκασίματα. Επίσης, η υλοποίηση βασίστηκε ελάχιστα στο σχετικό tutorial της βιβλιοθήκη xj3d, εφόσον τις περισσότερες φορές τα links δεν λειτουργούσαν, τα παραδείγματα ήταν ανεπαρκή ενώ δεν υπήρχε καν documentation για τις υπάρχουσες μεθόδους, γεγονός που μας ανάγκασε πολλές φορές να προχωράμε στα τυφλά και να ανακαλύπτουμε τις δυνατότητες μιας μεθόδου μέσα από πειραματισμούς. Ένα χαρακτηριστικό παράδειγμα που θα μπορούσαμε να είχαμε γλιτώσει χρόνο εάν υπήρχε σχετική ενημέρωση στο tutorial του xj3d ήταν η διαγραφή ενός αντικειμένου από τον κόσμο, που προϋποθέτει να είναι το τελευταίο που έχει εισαχθεί για να γίνει σωστά, εφόσον θεωρείται ο κόμβος ρίζα.

Παραταύτα, θεωρούμε πως ο βασικός στόχος της μεταπτυχιακής μας διατριβής, δηλαδή η δημιουργία μιας παραθυρικής εφαρμογής που θα σχεδιάζει έναν εικονικό κόσμο από μηδενική βάση, έχει επιτευχθεί. Η βελτιστοποίηση της εφαρμογής ήταν εξ αρχής γνωστό πως θα χρειαζόταν, εφόσον οι δυνατότητες που ανοίγονται στην περίπτωση σχεδίασης ενός εικονικού κόσμου είναι πολλές και καμιά φορά απρόβλεπτες, ενώ κάθε φορά θα μπορούσαμε να επανέλθουμε με μια καινούρια ιδέα υλοποίησης χωρίς να θέτουμε όριο στην υλοποίηση. Για αυτόν τον λόγο, η εφαρμογή κάλυψε ένα βασικό, πρωταρχικό κομμάτι λειτουργικότητας και σχεδιάστηκε με τρόπο ώστε να είναι επεξεργάσιμη και επεκτάσιμη μελλοντικά.

Σε κάθε περίπτωση, η εφαρμογή Reve World Designer είναι ένα πρωτοποριακό σχεδιαστικό εργαλείο, εφόσον αξιοποιεί το γράφημα σκηνής για την σύνθεση εικονικών κόσμων μέσω της αναπαράστασης REVE, που καθιστά εύκολη και προσιτή στον απλό χρήστη την σχεδίαση εικονικών κόσμων.

## Βιβλιογραφία

- [1] Loeffler, C. E. & Anderson, T. (eds) (1994). *The Virtual Reality Casebook*. Van Nostrand Reinhold. New York.
- [2] Ivan Sutherland *The Ultimate Display* (1965)
- [3] Krueger, M. W. (1991). *Artificial reality* (2nd ed.). Reading, MA: Addison-Wesley.
- [4] [https://el.wikipedia.org/wiki/%CE%95%CE%B9%CE%BA%CE%BF%CE%BD%CE%B9%CE%BA%CE%AE\\_%CF%80%CF%81%CE%B1%CE%B3%CE%BC%CE%B1%CF%84%CE%B9%CE%BA%CF%8C%CF%84%CE%B7%CF%84%CE%B1](https://el.wikipedia.org/wiki/%CE%95%CE%B9%CE%BA%CE%BF%CE%BD%CE%B9%CE%BA%CE%AE_%CF%80%CF%81%CE%B1%CE%B3%CE%BC%CE%B1%CF%84%CE%B9%CE%BA%CF%8C%CF%84%CE%B7%CF%84%CE%B1)
- [5] Bricken, W. (1990), *Learning in Virtual Reality*, Technical report No. HITL-M-90-5, University of Washington
- [6] Burdea, G., and P. Coiffet, *Virtual Reality Technology*, John Wiley & Sons, New York, USA, 1994
- [7] Beier, K. P. (2004). *Virtual reality: A short introduction*. Retrieved February 20, 2004
- [8] Thalmann, N, 2009. *Virtual Reality Software and Technology* & Howard, David, 2009. *First Virtual Reality Technology To Let You See, Hear, Smell, Taste And Touch*
- [9] Μπιλάλης Ν., Πετούσης Μ., Αντωνιάδης Α., «Εικονική Πραγματικότητα», Δελτίο Π.Σ.Δ.Μ.-Η., Μάρτιος 2001.
- [10] Χαρίτος Δ., Μαρτάκος Δ., *Εικονική Πραγματικότητα, Μεταπτυχιακό Πρόγραμμα Σπουδών, Β' Εξάμηνο, Τμήμα Πληροφορικής Πανεπιστημίου Αθηνών*, 1999.
- [11] Ellis, SR (1993) (ed) *Pictorial communication in virtual and real environments*
- [12] Dede, C, Salzman, M and Loftin, R B (1996) - 'The Development of a Virtual World for Learning Newtonian Mechanics'. In Brusilovsky, P, Kommers, P and Streitz, N (eds) *'Multimedia, Hypermedia and Virtual Reality'*. Springer Verlag Lecture Notes in Computer Science, pages 87-106
- [13] <https://www.javelinstrategy.com>
- [14] Aylett and Luck, 2000; Aylett and Cavazza, 2001 *Advancing Artificial Intelligence through Biological Process Applications*
- [15] Badler AGENTS '00 *Proceedings of the fourth international conference on Autonomous agents 1996* Pages 293-300
- [16] Mckenna, M.; Zeltzer, D. (MIT, Cambridge, MA (United States)) *Three dimensional visual display systems for virtual environments*
- [17] [http://www.it.uom.gr/project/mycomputer/v\\_card/perigr1.html](http://www.it.uom.gr/project/mycomputer/v_card/perigr1.html)
- [18] Γραφικά και Οπτικοποίηση. Αρχές και Αλγόριθμοι. Θ. Θεοχάρης, Γ. Παπαϊωάννου, Ν. Πλατής, Ν.Μ. Πατρικαλάκης
- [19] Appel, A. (1968). "Some techniques for shading machine renderings of solids" (PDF). *Proceedings of the Spring Joint Computer Conference* 32. pp. 37–49
- [20] Aaron E. Walsh, July 01, 2002 *Understanding Scene Graphs*
- [21] Bar-Zeev, Avi. "Scenographs: Past, Present, and Future"
- [22] Carey, Rikk and Bell, Gavin (1997). "The Annotated VRML 97 Reference Manual"
- [23] Russell S., Norvig P. ( 2005). 'Τεχνητή Νοημοσύνη, Μια σύγχρονη προσέγγιση', 2η Έκδοση, Εκδόσεις Κλειδάριθμος
- [24] *Composable Controllers for Physics-Based Character Animation*", Petros Faloutsos, Michiel van de Panne and Demetri Terzopoulos, Los Angeles, August, SIGGRAPH 2001, pp. 251-260
- [25] Gibet. *Intelligent Virtual Agents*, 432-438, 2009
- [26] *introduction to Agent-Based Modeling* Reynolds, 1987
- [27] Muse and Thalmann, 1997 *Integration of motion control techniques for virtual human and avatar real-time animation*

- [28] Design of Intelligent Multi-Agent Systems: (Funge et al., 1999)
- [29] Moffat, D. And Frijda, N.H. (1995) Where there's a will there's an agent, in M. Wooldridge and N.R. Jennings, Intelligent Agents: Theories, Architectures and Languages
- [30] Sloman, A. and Croucher, M. (1981). Why Robots Will Have Emotions. In Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI81), Vancouver, Canada.
- [31] Brooks, Rodney A. (1986), A robust layered control system for a mobile robot, IEEE Journal of Robotics and Automation 2:14-23.
- [32] Wooldridge and Jennings, 1995 Intelligent Agents: Theory and Practice (1995)
- [33] Shawver, D. (1997). Virtual actors and avatars in a flexible, user-determined-scenario environment. In Proceedings of the Virtual Reality Annual International Symposium, pages 1–5, Albuquerque, NM
- [34] CAVAZZA M., "An integrated parser for TFG with explicit tree typing" 1998
- [35] Realistic Avatars and Autonomous Virtual Humans Magnetat Thalmann and Thalmann, 1998 in VLNET Networked Virtual Environments
- [36] Jonathan Gratch, "Human-like behavior, alas, demands human-like intellect," Agents 2000
- [37] Schroeder The Social Life of Avatars: Presence and Interaction in Shared Virtual reality 2000
- [38] Hoffmam 1998 Virtual reality in the treatment of spider phobia: a controlled study
- [39] Second life Kelton A., J., 2008 Linden, Lab 2010
- [40] Using avatars and virtual environments in learning: What do they have to offer? Antonaccietal (2008)
- [41] Advanced Graphics Programming Techniques Using OpenGL. Tom McReynolds. 1998. Silicon Graphics – SIGGRAPH '98.
- [42] The Virtual Reality Modeling Language (VRML), Part 1: Functional specification and UTF-8 encoding (ISO/IEC 14772-1:1997). Web 3D Consortium. <http://www.web3d.org/technicalinfo/specifications/vrml97/>.
- [43] The Virtual Reality Modeling Language (VRML), Part 2: External authoring interface (ISO/IEC 14772-2:1997). Web 3D Consortium. <http://www.web3d.org/technicalinfo/specifications/eai/> .
- [44] <http://www.acm.org/tsc/vrml.html>
- [45] Extensible 3D (X3D), Part 1: Architecture and base components (ISO/IEC FDIS 19775-1:200x). Web 3D Consortium. <http://www.web3d.org/x3d.html>.
- [46] Extensible 3D (X3D), Part 2: Scene access interface (SAI) (ISO/IEC FDIS 19775-2:200x). Web 3D Consortium. <http://www.web3d.org/x3d/sai.html>.
- [47] Extensible 3D (X3D) encodings, Part 1: XML encoding (ISO/IEC FDIS 19776-1:200x). Web 3D Consortium. <http://www.web3d.org/x3d.html>.
- [48] <http://www.web3d.org/realtime-3d/x3d/profiles>
- [49] [http://en.wikipedia.org/wiki/Java\\_3D](http://en.wikipedia.org/wiki/Java_3D)<http://www.java3d.org/introduction.html>
- [50] <http://docs.unity3d.com/Manual/LearningtheInterface.html>
- [51] <http://www.autodesk.com/products/3ds-max/overview>
- [52] Anastassakis G., 2010, "Αναπαράσταση και Λειτουργία Εικονικών Περιβαλλόντων", PhD Thesis at University of Piraeus