

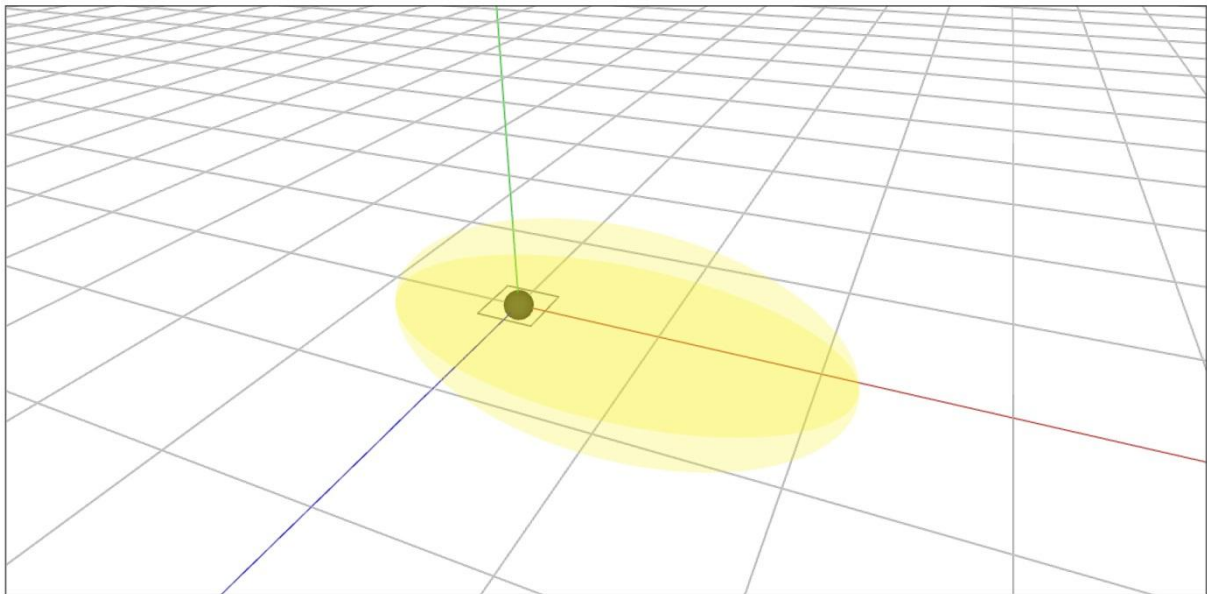


ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Τίτλος: Χωρικός ήχος σε περιβάλλον Χ3D

Ελένη Βονόρτα (Α.Μ. 3289)



Επιβλέπων καθηγητής: Μαλάμος Αθανάσιος  
Επιτροπή Αξιολόγησης : Μαλάμος Αθανάσιος  
Παχουλάκης Ιωάννης  
Παναγιωτάκης Σπύρος  
Ημερομηνία Παρουσίασης : 21/9/2016



## Abstract

Spatial sound has a conceptual role in the Web3D environments, due to highly realism scenes that can provide. Lately the efforts are concentrated on the extension of the X3D/X3DOM through spatial sound attributes. This paper presents a novel method for the introduction of spatial sound components in the X3DOM framework, based on X3D specification and Web Audio API. The proposed method incorporates the introduction of enhanced sound nodes for X3DOM which are derived by the implementation of the X3D standard components, enriched with accessional features of Web Audio API. Moreover, several examples-scenarios developed for the evaluation of our approach. The implemented examples established the achievability of new registered nodes in X3DOM, for spatial sound characteristics in Web3D virtual worlds.

## Σύνοψη

Ο Χωρικός ήχος έχει σημαντικό ρόλο στα Web3D περιβάλλοντα, λόγω των σκηνών υψηλού ρεαλισμού που παρέχει. Τελευταία έγιναν προσπάθειες εστιασμένες στην επέκταση του X3D/X3DOM μέσα από τα χαρακτηριστικά του χωρικού ήχου. Η δημοσίευση αυτή παρουσιάζει μια νέα μέθοδο για την εισαγωγή της σύνθεσης του χωρικού ήχου στο πλαίσιο του X3DOM, βάση της προδιαγραφής του X3D και του Web Audio API. Η προτεινόμενη μέθοδος ενσωματώνει την εισαγωγή των βελτιωμένων ηχητικών κόμβων (nodes) για το X3DOM που προέρχονται από την εκτέλεση των πρότυπων χαρακτηριστικών του X3D, εμπλουτισμένα με πρόσθετα χαρακτηριστικά από το Web Audio API. Επιπλέον, μερικά παραδείγματα-σενάρια αναπτύχθηκαν για την αξιολόγηση αυτής της προσέγγισης. Τα εκτελεσμένα παραδείγματα καθιέρωσαν την επίτευξη της εγγραφής νέων κόμβων στο X3DOM, με χαρακτηριστικά χωρικού ήχου σε Web3D εικονικούς κόσμους.

## Περιεχόμενα

Abstract .....	III
Σύνοψη .....	IV
Πίνακας Εικόνων .....	VI
1 Εισαγωγή .....	1
1.1 Περίληψη.....	1
1.2 Κίνητρο για την διεξαγωγή της εργασίας .....	2
1.3 Σκοπός και στόχοι εργασίας.....	2
1.4 Δομή εργασίας .....	3
2 Μεθοδολογία Υλοποίησης .....	4
2.1 Μέθοδος Ανάλυσης & Ανάπτυξης Πτυχιακή .....	4
2.2 Τεχνολογίες .....	4
2.2.1 HTML5 .....	4
2.2.2 JavaScript.....	5
2.2.3 X3D/ X3DOM .....	6
2.2.3.1 Web 3D - X3D/X3DOM .....	6
2.2.3.2 X3D/X3DOM Audio.....	6
2.2.4 Web Audio API.....	7
2.2.5 Audio .....	8
2.2.5.1 3D Audio.....	8
2.2.5.2 Web 3D Audio .....	9
3 Σχέδιο δράσης για την εκπόνηση της εργασίας.....	10
3.1 State of the Art .....	10
3.1.1 X3D .....	10
3.1.2 Web Audio API.....	15
3.1.2.1 Χωρικοποίηση (Spatialization/Panning) .....	15
3.1.2.2 Διεπαφή PannerNode .....	22
3.1.2.3 Διεπαφή AudioListener .....	29
3.1.2.4 Διεπαφή StereoPannerNode.....	32
3.2 Σημαντικοί στόχοι για την ολοκλήρωση της πτυχιακής.....	33
3.2.1 Χρονοδιάγραμμα Εργασίας (Gantt chart).....	34
4 Σύστημα εικονικής αναπαράστασης του ήχου σε 3D χώρο .....	35
4.1 Σχεδιασμός υλοποίησης.....	35
4.1.1 Web Audio API/ Εγγραφή ενισχυμένων X3D κόμβων στο X3DOM .....	35

4.1.2 Εγγεγραμμένοι κόμβοι του Web Audio API στο X3DOM .....	35
4.1.3 Ενισχυμένοι κόμβοι X3D .....	36
4.1.4 JavaScript Controller.....	37
4.2 Υλοποίηση .....	38
5 Αποτελέσματα.....	47
5.1 Συμπεράσματα .....	47
5.2 Μελλοντική εργασία και επεκτάσεις .....	47
Βιβλιογραφία .....	48

## Πίνακας Εικόνων

Εικόνα 1- Δομή του κόμβου του ήχου .....	10
Εικόνα 2- Γεωμετρία ηχητικού κόμβου.....	12
Εικόνα 3- Στερεοφωνικό panning .....	13
Εικόνα 4- Δομή του κόμβου AudioClip.....	14
Εικόνα 5- Διαδικασία του panning μιας πηγής χρησιμοποιώντας HRTF .....	20
Εικόνα 6- Αλγόριθμοι του DistanceModelType .....	24
Εικόνα 7- Παράμετροι του setOrientation της Διεπαφής PannerNode .....	26
Εικόνα 8- Παράμετροι του setPosition της Διεπαφής PannerNode .....	27
Εικόνα 9- Μέλη του PannerOptions.....	27
Εικόνα 10- Χαρακτηριστικά/ Ιδιότητες της Διεπαφής AudioListener.....	29
Εικόνα 11- Παράμετροι του setOrientation της Διεπαφής AudioListener .....	31
Εικόνα 12- Παράμετροι του setPosition της Διεπαφής AudioListener .....	31
Εικόνα 13- Χρονοδιάγραμμα Εργασίας (Gantt chart).....	34
Εικόνα 14- Σχεδιασμός θέσεων του κινηματογράφου στο 3DS Max .....	39
Εικόνα 15- Σχεδιασμός αντικειμένων του κινηματογράφου στο 3DS Max .....	39
Εικόνα 16- Δεξιά όψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max .....	40
Εικόνα 17- Πάνω-εμπρόσθια όψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max .....	40
Εικόνα 18- Κάτωψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max .....	41
Εικόνα 19- Εξαγωγή σκηνής σε XHTML με τη χρήση του InstantExport.....	42
Εικόνα 20- Προσδιορισμός ηχητικών πηγών στη σκηνή .....	43
Εικόνα 21- Κώδικας σύνδεσης του ήχου στην σκηνή X3DOM.....	44
Εικόνα 22- Transform group ενός ηχείου της σκηνής.....	44
Εικόνα 23- Οπτική γωνία πολυθρόνας κεντρικού τμήματος .....	45
Εικόνα 24- Οπτική γωνία πολυθρόνας δεξιού τμήματος .....	45
Εικόνα 25- Γενική οπτική γωνία της σκηνής του κινηματογράφου .....	46

# 1 Εισαγωγή

## 1.1 Περίληψη

Η τρισδιάστατη (3D) απεικόνιση παίζει ζωτικό ρόλο στα γραφικά υπολογιστών. Τα τελευταία χρόνια, υπάρχει ένα ανεπτυγμένο ενδιαφέρον στις τεχνολογίες που έχουν σχεδιαστεί για την παρουσίαση τρισδιάστατων σκηνών στο διαδίκτυο. Αυτή η προσπάθεια ξεκίνησε από το 1995, όταν μια κειμενοστραφής μετα-γλώσσα, η Γλώσσα Μοντελοποίησης Εικονικής Πραγματικότητας (Virtual Reality Modeling Language – VRML), σχεδιάστηκε από την επίδραση της HTML. [1]

Υστερα, η Βιβλιοθήκη Διαδικτυακών Γραφικών (Web Graphics Library – WebGL) παρουσιάστηκε το 2011 από την Kronos Group [2] και προσέλκυσε ιδιαίτερα την προσοχή του T.N. Eicke [3], λόγω της αποτελεσματικής προτεινόμενης μεθόδου αναπαράστασης διαδικτυακών τρισδιάστατων απεικονίσεων, χωρίς συγκεκριμένες απαιτήσεις από εξοπλισμό και λογισμικό. Το επονομαζόμενο Επεκτάσιμο 3D (Extensible 3D – X3D), είναι μία από τις τελευταίες αρχιτεκτονικές που προέκυψε σαν επέκταση της VRML [4]. Η χρήση της Επεκτάσιμης Γλώσσας Χαρακτηρισμού (Extensible Markup Language – XML) στο X3D είχε ως συνέπεια την αύξηση της ελαστικότητας και την πλήρη ανάπτυξη του σε σχέση με την VRML [5].

Ωστόσο, το μειονέκτημα για εγκατάσταση προεκτάσεων παρέμενε. Το πρόβλημα επιλύθηκε από την πρόταση του πλαισίου X3DOM, όπως αναφέρθηκε από τον J. Behr [6]. Γι' αυτό το λόγο, τα τελευταία πέντε χρόνια περίπου το 70% της προδιαγραφής του X3D υλοποιήθηκε στο πλαίσιο X3DOM, αλλά τα χαρακτηριστικά του χωρικού ήχου παραμένουν ανοικτό θέμα και δεν έχουν ακόμα υλοποιηθεί, παρά το γεγονός ότι είναι ένα σημαντικό θέμα για την 3D σκηνή.

Επιπλέον, η βιβλιογραφία για τον 3D ήχο δείχνει ότι έχει προσελκύσει περισσότερη προσοχή από ερευνητικές ομάδες. Οι ομάδες των K. Garbe [7] και H. Ding [8] μέσα από την ενίσχυση των δικών τους X3D ηχητικών κόμβων (sound nodes), κατάφεραν μια καλύτερη αναπαράσταση διαδικτυακών τρισδιάστατων απεικονίσεων. Εν τω μεταξύ, η πιο ενδιαφέρουσα προσέγγιση σε αυτό το θέμα προτάθηκε από το Web Audio API. Το οποίο περιγράφει ένα JavaScript API [9] υψηλού επιπέδου που παρέχει μια ομάδα από νέες και ενισχυμένες ιδιότητες ήχου στις διαδικτυακές εφαρμογές [10].

Οι Wyse [11] και Pettersson [12] ανέπτυξαν μια νέα υλοποίηση χρησιμοποιώντας το Web Audio API με σκοπό να επιτύχουν τους στόχους τους για χωρικό ήχο. Οι περισσότερες 3D εφαρμογές υποστηρίζουν χωρικό ήχο. Για παράδειγμα, η θέση και η ένταση μιας πηγής ήχου θα μπορούσε να ελέγχεται από τον χρήστη και αναλόγως, θα μπορούσε να αλλάζει εξαρτώμενη από την κίνηση του χρήστη. Όλα τα παραπάνω είναι σημαντικά για μια ρεαλιστική σκηνή. Επομένως, ο χωρικός ήχος είναι εξίσου καθοριστικός για τα 3D αλλά και για τις διαδικτυακές 3D εφαρμογές.

Ωστόσο, ενώ το X3D ISO [13] περιλαμβάνει προδιαγραφή για τη σύνθεση χωρικού ήχου, δεν έχει υλοποιηθεί σε X3DOM ακόμα. Επιπλέον, δεν υπάρχει αποτελεσματική υλοποίηση του χωρικού ήχου σε X3D γενικά. Εξάλλου, μερικές δημοσιεύσεις είναι διαθέσιμες στη βιβλιογραφία που απευθύνεται στο θέμα της επέκτασης του X3D ήχου [7] [14] αλλά δουλεύουν μόνο με περιορισμένο αριθμό προηγμένων χαρακτηριστικών και μεθόδων του ήχου. Συνεπώς, παρουσιάζουμε μια καινοτόμα μέθοδο για τη βελτίωση της χωρικής ακουστικότητας στο X3DOM, μέσω της προδιαγραφής του X3D και της δομής του Web Audio API. Λεπτομερώς, κάνουμε προσπάθεια να συνδυάσουμε τα οφέλη του X3DOM, όπως είναι η ανοιχτή τεχνολογία και η έλλειψη της χρήσης προσθέτων, με την ευέλικτη HTML5 και το αποτελεσματικό Web Audio API.

Επομένως, η δουλειά μας σκοπεύει να συμπεριλάβει τον 3D ήχο σε μια δηλωτική διαδικτυακή 3D σκηνή, με την ικανότητα να τρέχει τοπικά. Επίσης, η δημοσίευση αυτή δεν επικεντρώνεται μόνο στην υλοποίηση αλλά και στην αξιολόγηση μέσα από ένα πλήθος σεναρίων.

## 1.2 Κίνητρο για την διεξαγωγή της εργασίας

Το πλαίσιο X3DOM περιλαμβάνει θεμελιώδη ρόλο στην ανάπτυξη 3D στο web, διότι παρέχει μια προσέγγιση για την ενσωμάτωση των Declarative 3D σε HTML5 [19]. Ωστόσο, η εφαρμογή του χωρικού ήχου στο X3DOM εξακολουθεί να είναι ανεπαρκής, παρόλο που ο χωρικός ήχος θα πρέπει να αποτελεί αναπόσπαστο μέρος μιας 3D εφαρμογής. Για να ξεπεραστεί αυτό το μειονέκτημα, παρουσιάζουμε μια καινοτόμο λύση του χωρικού ήχου στο πλαίσιο του X3DOM που βασίζεται σε μια συνδυαστική μεθοδολογία.

Συγκεκριμένα, προτείνουμε τον εμπλουτισμό του X3DOM με χαρακτηριστικά χωρικού ήχου, χρησιμοποιώντας και τους ηχητικούς κόμβους του X3D και τη δομή του Web Audio API. Έχουμε επιλέξει αυτό το συνδυασμό λόγω του ότι τόσο το X3D αλλά και το Web Audio API έχουν επηρεαστεί από το OpenAL. Ιδιαίτερα, το Web Audio API έχει δανειστεί πολλές έννοιες από το OpenAL (θέση και προσανατολισμό των πηγών και των ακροατών, οι παράμετροι που σχετίζονται με τους κώνους της πηγής ήχου, σχετικές ταχύτητες των πηγών και των ακροατών) [29]. Ταυτόχρονα, η δομή του OpenAL έχει χρησιμοποιηθεί για επέκταση των ηχητικών κόμβων του X3D, από την ομάδα της Garbe [7] και αρκετοί συγγραφείς [54], [55], [18] έχουν προτείνει την εφαρμογή του 3D εικονικού περιβάλλοντος με τη συνεργασία των X3D (για την 3D σκηνή) και του OpenAL (για τον 3D ήχο).

Αν και το Web Audio API έχει συνδυαστεί με το WebGL για το χωρικό ήχο στο παρελθόν [36], οι προσπάθειες αυτές είναι προσαρμοσμένες υλοποιήσεις που χρησιμοποιούν έμφυτες μεθόδους του API και είναι ακόμα μακριά από να γίνει μια πλατφόρμα ή ακόμα περισσότερο μια γλώσσα. Η προσπάθειά μας δεν ήταν απλά να προσαρμόσουμε ένα περιβάλλον, αλλά αντίθετα να ενσωματωθούν πλήρως το Web Audio API σε X3D γλώσσα, με την εισαγωγή νέων ηχητικών κόμβων και την ενσωμάτωσή τους σε X3DOM API. Με άλλα λόγια, με αυτή την προσέγγιση ο πολύπλοκος σχεδιασμός και η εφαρμογή ηχητικών γίνεται διαφανής στον προγραμματιστή και επιπλέον οι εφαρμογές είναι ανεξάρτητες από τις ηχητικές βιβλιοθήκες που χρησιμοποιούνται.

Το Σχ. 3 παρουσιάζει τους νέους κόμβους που έχουν καταχωρηθεί στο X3DOM σύμφωνα με την πρότασή μας (ένας συνδυασμός από κόμβους του Web Audio API και προσαρμοσμένους κόμβους). Ο Πίνακας 1, Σχ. 2α και Σχ. 2β απεικονίζουν την αντιστοίχιση των χαρακτηριστικών του X3D και του Web Audio API, τα οποία έχουν ενσωματωθεί ώστε να δημιουργήσουμε τις αντίστοιχες συνιστώσες στο X3DOM.

## 1.3 Σκοπός και στόχοι εργασίας

Ο στόχος αυτής της εργασίας είναι η ενσωμάτωση του χωρικού ήχου σε διαδικτυακές τρισδιάστατες σκηνές σε περιβάλλον X3DOM. Για την επίτευξη αυτού του στόχου χρησιμοποιήσαμε τη δομή του Web Audio API σε συνδυασμό με τους ηχητικούς κόμβους του X3D. Η συμβατότητα η οποία πηγάζει από το κοινό έδαφος που έχουν το API και το X3D, είναι το κύριο πλεονέκτημα που εκμεταλλευόμαστε προκειμένου να ενσωματωθεί ο χωρικός ήχος στο πλαίσιο του X3DOM. Μια μέτρηση του αντίκτυπου της συνεισφοράς μας είναι ο αριθμός των web εφαρμογών στην πλατφόρμα του X3DOM που μπορούν να επωφεληθούν τα στοιχεία του χωρικού ήχου που εισάγουμε. Τηλεδιάσκεψη, παιχνίδια, καθηλωτικό internet και ψυχαγωγία είναι μερικές από τις περιοχές που επιτυγχάνουν μια πραγματικά εκπληκτική εμπειρία ακρόασης, μέσω του 3D ήχου του X3DOM.



## 1.4 Δομή εργασίας

Σε αυτή την ενότητα θα κάνουμε μια σύντομη παρουσίαση των περιεχομένων κάθε κεφαλαίου της εργασίας.

Στο πρώτο κεφάλαιο κάνουμε μια εισαγωγή για το X3D, την εξέλιξή του στο X3DOM, τις δυνατότητες του όσον αφορά τον ήχο και την χρήση του Web Audio API ώστε να προσθέσουμε την δυνατότητα του χωρικού ήχου. Αναφέρουμε επίσης τα κίνητρα για την διεξαγωγή της εργασίας αλλά και τους στόχους και σκοπούς για τους οποίους υλοποιήθηκε.

Στο δεύτερο κεφάλαιο αναφέρουμε την μεθοδολογία υλοποίησης της εργασίας. Γίνεται μια αναφορά στις τεχνολογίες που χρησιμοποιήθηκαν ώστε να επιτύχουμε το αποτέλεσμα του χωρικού ήχου.

Στο τρίτο κεφάλαιο γίνεται αναφορά στην βιβλιογραφική αναζήτηση της τεχνολογίας αιχμής (state of the art) και την υλοποίηση του σχεδίου δράσης για την εκπόνηση της πτυχιακής εργασίας.

Στο τέταρτο κεφάλαιο περιλαμβάνει το κύριο μέρος της πτυχιακής. Γίνεται ανάλυση του προβλήματος του ήχου στις διαδικτυακές εφαρμογές και παρουσιάζεται ο σχεδιασμός υλοποίησης του χωρικού ήχου στο περιβάλλον X3DOM.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα αποτελέσματα της εργασίας και γίνεται αναφορά για μελλοντικές εργασίες και επεκτάσεις πάνω στο θέμα του διαδικτυακού ήχου.

## 2 Μεθοδολογία Υλοποίησης

### 2.1 Μέθοδος Ανάλυσης & Ανάπτυξης Πτυχιακή

Για την υλοποίηση αυτής της εργασίας έγινε μελέτη του Web Audio API αλλά και των δυνατοτήτων του X3DOM στο κομμάτι του ήχου με σκοπό την ενσωμάτωση του χωρικού ήχου στο περιβάλλον του X3DOM. Λόγω των κοινών χαρακτηριστικών ανάμεσα στο Web Audio API και το X3DOM, επιτεύχθηκε η δημιουργία δυο νέων κόμβων (AudioSound και AudioSource) τους οποίους χρησιμοποιούμε στην υλοποίηση ενός παραδείγματος ώστε να δείξουμε την ικανότητα αναπαράστασης χωρικού ήχου.

Στις παρακάτω ενότητες θα γίνει αναφορά τόσο στο Web Audio API αλλά και το X3DOM όσον αφορά τον χωρικό ήχο. Θα εξεταστούν οι κόμβοι και οι διεπαφές των δυο τεχνολογιών και θα αναλυθεί η υλοποίηση που έγινε για την ενσωμάτωση του χωρικού ήχου.

Για ότι αφορά το οπτικό μέρος του παραδείγματος αυτής της πτυχιακής έγινε χρήση του Autodesk 3DS Max και με τη χρήση του plugin InstantExport για εξαγωγή της 3D σκηνής σε μορφή XHTML. Για την σωστή αναπαράσταση των ηχείων στο συγκεκριμένο παράδειγμα έγινε μελέτη της τεχνολογίας surround Dolby Atmos.

Με την ολοκλήρωση αυτής της πτυχιακής γίνεται μια παρουσίαση των αποτελεσμάτων της υλοποίησης και μια αναφορά για μελλοντικές εξελίξεις.

### 2.2 Τεχνολογίες

Σε αυτή την ενότητα θα αναλύσουμε τις τεχνολογίες που χρησιμοποιήθηκαν για την εκπόνηση αυτής της εργασίας. Γίνεται μια αναφορά στον 3D ήχο και τη σημαντικότητά του στους εικονικούς κόσμους αλλά και στις διαδραστικές 3D εφαρμογές. Αναφέρονται επίσης οι δυνατότητες του X3D και του Web Audio API σχετικά με το θέμα του ήχου και μερικά παραδείγματα βελτίωσης στο περιβάλλον X3DOM.

#### 2.2.1 HTML5

Η HTML5 (HyperText Markup Language) είναι μια γλώσσα σήμανσης που χρησιμοποιείται για τη δομή και την παρουσίαση περιεχομένου στο διαδίκτυο. Είναι στην πέμπτη έκδοση του στάνταρ της HTML.

Αυτή η έκδοση δημοσιοποιήθηκε τον Οκτώβριο του 2014 από το World Wide Web Consortium (W3C) ώστε να βελτιώσει τη γλώσσα για την υποστήριξη των τελευταίων πολυμεσικών περιεχομένων, καθιστώντας την παράλληλα ευανάγνωστη για τον άνθρωπο και σταθερά κατανοητή από τους υπολογιστές και τις συσκευές όπως επίσης και από τους περιηγητές, τους parsers κλπ. Η HTML5 προορίζεται να συμπεριλάβει όχι μόνο την HTML 4, αλλά επίσης την XHTML 1 και το DOM επιπέδου 2 της HTML.

Η HTML5 περιλαμβάνει λεπτομερή μοντέλα επεξεργασίας για να ενθαρρύνει περισσότερο διαλειτουργικές εφαρμογές· επεκτείνει, βελτιώνει και αιτιολογεί τη σήμανση που διατίθενται για τα έγγραφα, και εισάγει σήμανση και διεπαφές προγραμματισμού εφαρμογών (API) για σύνθετες

εφαρμογές web. Για τους ίδιους λόγους, η HTML5 είναι επίσης υποψήφια για εφαρμογές κινητών cross-platform, γιατί περιλαμβάνει υπόψη λειτουργίες που έχουν σχεδιαστεί με χαμηλής-ισχύος συσκευές.

Περιλαμβάνονται πολλά νέα συντακτικά χαρακτηριστικά. Για να περιλαμβάνει τοπικά και να χειρίζεται το περιεχόμενο πολυμέσων και γραφικών, προστέθηκαν τα νέα στοιχεία (elements) <video>, <audio> και <canvas>, και υποστηρίζει περιεχόμενο για Κλιμακούμενα Διανυσματικά Γραφικά (Scalable Vector Graphics - SVG) και MathML για μαθηματικούς τύπους. Για να εμπλουτιστεί το σημασιολογικό περιεχόμενο των εγγράφων, έχουν προστεθεί νέα στοιχεία για τη δομή της σελίδας όπως τα <main>, <section>, <article>, <header>, <footer>, <aside>, <nav> και <figure>. Έχουν εισαχθεί νέα χαρακτηριστικά (attributes), μερικά στοιχεία και χαρακτηριστικά έχουν αφαιρεθεί, και άλλα όπως τα <a>, <cite> και <menu> έχουν τροποποιηθεί, επαναπροσδιοριστεί ή τυποποιηθεί.

Τα APIs και το Document Object Model (DOM) είναι πλέον θεμελιώδη τμήματα της HTML5 προδιαγραφής και της HTML5, επίσης ορίζει καλύτερα την επεξεργασία των μη έγκυρων εγγράφων. [\[59\]](#)

## 2.2.2 JavaScript

Η JavaScript είναι μια υψηλού επιπέδου, δυναμική και ερμηνευτική γλώσσα προγραμματισμού. Έχει τυποποιηθεί στην προδιαγραφή της γλώσσας ECMAScript. Παράλληλα με την HTML και το CSS, είναι μία από τις τρεις βασικές τεχνολογίες για την παραγωγή περιεχομένου στον παγκόσμιο ιστό· η πλειοψηφία των ιστοσελίδων που το χρησιμοποιούν και υποστηρίζεται από όλους τους σύγχρονους περιηγητές, χωρίς πρόσθετα.

Η JavaScript βασίζεται σε πρότυπο με συναρτήσεις first-class που σημαίνει ότι οι συναρτήσεις είναι αντικείμενα και έχουν τύπο και συμπεριφορά, καθιστώντας την γλώσσα που υποστηρίζει περισσότερα από ένα προγραμματιστικά παραδείγματα αλλά επίσης υποστηρίζει αντικειμενοστραφή, επιτακτικά και λειτουργικά στυλ προγραμματισμού. Έχει ένα API για την χρήση κειμένου, πινάκων, ημερομηνιών και τις κανονικές εκφράσεις, αλλά δεν συμπεριλαμβάνει κανένα I/O, όπως δικτύωση, αποθήκευση ή λειτουργίες γραφικών. Βασίζεται για αυτά από το περιβάλλον υποδοχής στο οποίο είναι ενσωματωμένη.

Παρά το γεγονός ότι υπάρχουν ισχυρές εξωτερικά ομοιότητες μεταξύ JavaScript και Java, συμπεριλαμβανομένου του ονόματος της γλώσσας, τη σύνταξη, και τις αντίστοιχες στάνταρ βιβλιοθήκες, οι δύο γλώσσες είναι διαφορετικές και διαφέρουν σημαντικά στο σχεδιασμό τους. Η JavaScript επηρεάστηκε από γλώσσες προγραμματισμού, όπως η Self και η Scheme.

Η JavaScript χρησιμοποιείται επίσης σε περιβάλλοντα που δεν έχουν πρόσβαση στο διαδίκτυο, όπως είναι τα έγγραφα PDF, σε site-specific browsers (SSB), και σε γραφικά στοιχεία επιφάνειας εργασίας (desktop widgets). Οι νέες και πιο γρήγορες εικονικές μηχανές (VMs) με JavaScript και οι πλατφόρμες που χτίστηκαν πάνω τους έχουν επίσης αυξήσει τη δημοτικότητα της JavaScript για τις διαδικτυακές εφαρμογές server-side. Από την πλευρά του client, η JavaScript έχει παραδοσιακά εφαρμοστεί ως μια ερμηνευμένη γλώσσα, αλλά τα πιο πρόσφατα προγράμματα περιήγησης εκτελούν μεταγλώττιση just-in-time. Χρησιμοποιείται επίσης στην ανάπτυξη παιχνιδιών, τη δημιουργία επιτραπέζιων και κινητών εφαρμογών, καθώς και τον προγραμματισμό του δικτύου server-side με run-time περιβάλλοντα, όπως το Node.js. [\[61\]](#)

## 2.2.3 X3D/ X3DOM

### 2.2.3.1 Web 3D - X3D/X3DOM

Το X3D προτάθηκε από το Web3D Consortium, μια μη κερδοσκοπική οργάνωση η οποία είναι ένας συνδυασμός από επιχειρηματικές εταιρείες, κυβερνητικούς οργανισμούς, ακαδημαϊκά ιδρύματα, και ελεύθερους επαγγελματίες. Έχει εγκριθεί επίσημα από το Διεθνή Οργανισμό Τυποποίησης (ISO) ως το πρότυπο ISO/IEC 19775, από το 2004 [15]. Το X3D είναι σύγχρονος απόγονος της VRML, περιλαμβάνει όχι μόνο τις δυνατότητες της VRML, αλλά κυριαρχεί και σε διάφορες πτυχές. Ειδικότερα, χρησιμοποιεί την XML για να εκφράσει τη γεωμετρία και την ενσωματώνει απλά με άλλες εφαρμογές [16]. Επιπλέον, είναι οργανωμένο σε λογικές ομάδες λειτουργικότητας (components) [17], τα οποία θα μπορούσαν να επεκταθούν και να εμπλουτιστούν με νέα. Αντίστοιχα, οι εφαρμογές X3D είναι αξιόπιστες και προβλέψιμες, ομοίως η δυαδική μορφή του X3D παρέχει κρυπτογράφηση και συμπίεση [18]. Για τους παραπάνω λόγους, το X3D έχει γίνει πιο ελκυστικό και αποτελεσματικό από την VRML.

Από τη μία πλευρά, το X3D έχει ήδη χρησιμοποιηθεί σε HTML5, προκειμένου να δηλώσουμε τους κόσμους web 3D. Από την άλλη πλευρά, δεν αντιμετωπίζει το πρόβλημα της σύνδεσης μεταξύ του web-browser στο προσκήνιο και του X3D στο παρασκήνιο. Το X3DOM έρχεται για να ξεπεραστεί αυτό το πρόβλημα [19]. Είναι ένα πλαίσιο, το οποίο ενσωματώνει ένα σύνολο από τεχνολογίες όπως το X3D, WebGL, HTML, CSS και JavaScript. Ο κύριος στόχος είναι η ανάπτυξη μιας σκηνής X3D με τη χρήση της HTML DOM και τη διαχείριση του 3D περιεχομένου μέσα από τα στοιχεία DOM χωρίς επιπλέον πρόσθετα [20].

Εξίσου σημαντική είναι η ικανότητα του X3DOM, ως πλαίσιο JavaScript, για να επιτρέψει τρεις αλληλεπιδράσεις, η πρώτη είναι ένα γεγονός στη σκηνή που προκαλεί μια συμπεριφορά στη σκηνή. Η δεύτερη είναι ένα γεγονός στη σκηνή που προκαλεί μια συμπεριφορά στην HTML5. Η τελευταία είναι ένα γεγονός στην HTML5 που προκαλεί μια συμπεριφορά στην σκηνή [21]. Οι παρούσες ερευνητικές προσπάθειες επικεντρώνονται στην ανάπτυξη διαδραστικών εικονικών κόσμων του Web 3D, με βάση το X3DOM. Ωστόσο, ορισμένες μελέτες εφιστούν την προσοχή στην επέκταση του X3DOM κόμβου. Ο Σταμούλιας [22] προτείνει την υλοποίηση του στοιχείου της φυσικής σε άκαμπτο σώμα στο περιβάλλον X3DOM. Μια ιδέα για τη βελτίωση της αναπαράστασης των σκηνών για το X3DOM παρέχεται από την ομάδα του Kuipers [3]. Επιπλέον, ο Καπετανάκης [23] παρουσιάζει μια προσέγγιση για την επέκταση των μεθόδων προσαρμογής του X3DOM προσθέτοντας ένα μηχανισμό για να εκτελέσει δυναμική προσαρμογή και την επίτευξη διανομής HD βίντεο σε 3D κόσμους εικονικής πραγματικότητας (VR).

Όλα τα παραπάνω έχουν τον ίδιο στόχο, να αυξηθεί το επίπεδο του ρεαλισμού στην Web 3D σκηνή, μέσω της ενίσχυσης της δομής X3DOM. Ο στόχος της παρούσας εργασίας, σύμφωνα με τα προηγούμενα παραδείγματα, είναι να ανοίξει το πεδίο του χωρικού ήχου σε μια web 3D σκηνή, πέρα από τα όρια της τεχνολογίας του X3D-X3DOM.

### 2.2.3.2 X3D/X3DOM Audio

Η τρέχουσα βιβλιογραφία δείχνει ότι έχουν γίνει ανεπαρκείς προσπάθειες προκειμένου να ενσωματωθεί ο χωρικός ήχος στο X3D. Ιδιαίτερα, δύο τύποι κόμβων περιλαμβάνονται στο X3D, ο πρώτος είναι για την περιγραφή του ήχου και ο άλλος για την πηγή του ήχου.

Συγκεκριμένα, ο πρώτος κόμβος είναι ο X3DSoundNode, ο οποίος είναι ένας αφηρημένος κόμβος για όλους τους κόμβους του ήχου. Είναι μινιμαλιστικός με ένα μόνο χαρακτηριστικό, τα μεταδεδομένα, τα οποία εκφράζουν σημαντικές πληροφορίες για τη σημασία, την εμφάνιση και τον προτεινόμενο ρόλο του μοντέλου [51]. Ο δεύτερος κόμβος είναι ο X3DSoundSourceNode, ο οποίος

είναι ένας αφηρημένος κόμβος για κάθε κόμβο που χρησιμοποιείται για να εκπέμπει ήχο και έχει μια σειρά από κοινά πεδία με τον `TimeSensor`, για παράδειγμα `loop`, `startTime`, `stopTime`, `pauseTime` και `resumeTime`. Ο τρίτος κόμβος είναι ο ήχος, ο οποίος προέρχεται από τον `X3DSoundNode`. Έχει σχεδιαστεί για την περιγραφή των ήχων στην X3D σκηνή. Συγκεκριμένα, καθορίζει τόσο τη θέση και τη συμπεριφορά του ήχου.

Επιπλέον, η γεωμετρία περιγράφει ότι ο ήχος μπορεί να κατευθύνεται και να εκπέμπεται σε μια ελλειπτική τροχιά. Δύο ελλειψοειδή αποτελούν το μοτίβο, το οποίο καθορίζει τα όρια για το επίπεδο της έντασης του ήχου. Επίσης, τα ελλειψοειδή μπορούν να αναμορφωθούν με σκοπό την παροχή περισσότερης ή λιγότερης κατευθυντήριας εστίασης από τη θέση του ήχου [52]. Κατά συνέπεια, ο κόμβος του ήχου έχει σκοπό να αναγνωρίσει την πηγή και έχει σχέση με την κατεύθυνση, την τοποθεσία, την προτεραιότητα και γενικά, τα χωρικά χαρακτηριστικά της πηγής ήχου [53]. Ο τέταρτος κόμβος είναι ο `AudioClip`, ο οποίος προέρχεται από τον `X3DSoundSourceNode`. Προσδιορίζει τα δεδομένα του ήχου που μπορεί να αναφέρονται από τους κόμβους ήχου. Βασικά, φορτώνει ένα εξωτερικό αρχείο ήχου με σκοπό να χειριστεί την αναπαραγωγή, τη διακοπή και την εκκίνηση.

Όσον αφορά τα χαρακτηριστικά, το `AudioClip` έχει μια σειρά από πεδία κοινά με τον `TimeSensor`, επειδή είναι ένα `X3DSoundSourceNode` και υλοποιεί τον αφηρημένο τύπο του `X3DTimeDependentNode`. Βασικά, τα πεδία των ηχητικών κόμβων και η διαπλοκή τους παρουσιάζονται στο Σχήμα 1 από ένα ερμηνευτικό διάγραμμα. Εκτός αυτού, το X3DOM είναι απόγονος του X3D, το μόνο πράγμα που έχει εφαρμοστεί για τον ήχο είναι η καταγραφή των `X3DSoundNode`, `X3DSoundSourceNode`, `AudioClip` και των ηχητικών κόμβων, αλλά χωρίς τις περισσότερες από τις ιδιότητες των αντίστοιχων κόμβων X3D.

Συγκεκριμένα, μια σκηνή X3DOM θα μπορούσε να περιλαμβάνει ένα αρχείο ήχου για αναπαραγωγή, χωρίς χωροταξικά χαρακτηριστικά. Μόνο το χαρακτηριστικό “`SFBool enabled`” έχει επιπλέον προστεθεί στο X3DOM σε σύγκριση με το X3D, το οποίο καθορίζει εάν το κλιπ είναι ενεργοποιημένο ή όχι. Οι δύο πρώτες στήλες του Πίνακα 1 απεικονίζουν τα χαρακτηριστικά των ηχητικών κόμβων που μεταφέρονται στο X3DOM από το X3D.

## 2.2.4 Web Audio API

Το Web Audio API είναι ένα Javascript API υψηλού επιπέδου, το οποίο μπορεί να χρησιμοποιηθεί για την σύνθεση ήχου σε διαδικτυακές εφαρμογές. Επιπλέον, χαρακτηρίζεται ως ένα εξαιρετικά ισχυρό εργαλείο για τον έλεγχο του ήχου στο πρόγραμμα περιήγησης και τείνει να γίνει στην πραγματικότητα ένα πρότυπο στα σύγχρονα προγράμματα περιήγησης [33]. Μέχρι τώρα, οι μηχανές απόδοσης 3D ήχου αξιοποιήθηκαν, όπως είναι η OpenAL (Open Audio Library) [34], η βιβλιοθήκη FMOD, με σκοπό την ανάπτυξη τρισδιάστατης ακουστικότητας. Ακόμη πίστευαν ότι οι υπάρχουσες λύσεις παρέχουν φιλικές διεπαφές κι έχουν μια σειρά από περιορισμούς. Δηλαδή, θα περιορίζεται στην απόδοση των στατικών πηγών ήχου, σε αντίθεση με το Web Audio API [35].

Επιπλέον, η σπουδαιότητα του Web Audio API μπορεί να είναι εμφανής από το γεγονός ότι έχει αναπτυχθεί μια σειρά από διαδικτυακές βιβλιοθήκες ήχου και APIs, προκειμένου να χρησιμοποιηθούν. Συγκεκριμένα, το `Three.js` χρησιμοποιεί το Web Audio API για να παίξει τον ήχο και να καθορίσει τη σωστή ένταση. Πρόκειται για μια βιβλιοθήκη JavaScript, η οποία προσφέρει έναν απλό τρόπο προγραμματισμού WebGL απευθείας από το JavaScript, με σκοπό τη δημιουργία και τη ζωντανία των τρισδιάστατων σκηνών [36], [37]. Η βιβλιοθήκη `webaudiox.js` διαθέτει μια σειρά από βοηθούς για τη χρήση του Web Audio API και η βιβλιοθήκη `howler.js` υποστηρίζει αυτόματη προσωρινή αποθήκευση για το Web Audio API [38], [39]. Επίσης, το `pedalboard.js` είναι ένα πλαίσιο Javascript ανοιχτού κώδικα που αναπτύσσει ηχητικά εφέ μέσα από το Web Audio API [40].

Άλλη μια βιβλιοθήκη είναι η `Wad` που βοηθά στην απλούστευση του χειρισμού του ήχου χρησιμοποιώντας το Web Audio API [41]. Τέλος, η Javascript βιβλιοθήκη `Fifer` είναι ένας ελαφρύς οδηγός για το Web Audio API με εφεδρικό το Flash [42]. Όπως αναφέρθηκε παραπάνω, το Web

Audio API καθίσταται ελκυστικό για το χωρικό ήχο σε διαδικτυακά περιβάλλοντα. Εκτός από τους προηγούμενους λόγους, διακρίνεται επίσης για επιπλέον οφέλη. Πιο αναλυτικά, είναι ανοικτού κώδικα και υποστηρίζεται από τα περισσότερα προγράμματα περιήγησης. Επιπλέον, ο πολυκάναλος ήχος είναι διαθέσιμος και είναι ενσωματωμένος με το Web Real-Time Communications (WebRTC).

Επίσης, προσφέρονται υψηλού επιπέδου ηχητικές ικανότητες, όπως τα φίλτρα, οι γραμμές καθυστέρησης, οι ενισχυτές, τα εφέ χώρου (όπως panning). Ταυτόχρονα, τα κανάλια ήχου μπορούν να έχουν 3D διανομή ανάλογα με τη θέση, την ταχύτητα ή την κατεύθυνση του θεατή και της ηχητικής πηγής [43]. Επιπλέον, το Web Audio API χαρακτηρίζεται από συνθετικότητα, χρησιμοποιώντας μια δομή ηχητικού κόμβου η οποία μπορεί να συνδεθεί από κοινού προκειμένου να σχηματίσει ένα γράφημα δρομολόγησης του ήχου. Πέραν αυτού είναι πολύ πιο γρήγορο, δεδομένου ότι είναι γραμμένο σε C ++, αντί να ήταν γραμμένο σε JavaScript. Ωστόσο, το Web Audio API παρέχει επίσης έναν κόμβο (ScriptProcessorNode) που επιτρέπει στους web developers να χειριστούν τον ήχο χρησιμοποιώντας JavaScript.

Όλα τα παραπάνω στοιχεία κάνουν το Web Audio API να ανοίξει νέους δρόμους για τη σύνθεση web ήχου. Πράγματι, δεν ήταν τυχαίο το γεγονός ότι το Web Audio API έχει εστιάσει πολλή προσοχή από ερευνητικές ομάδες τα τελευταία δύο χρόνια. Ειδικότερα, η βιβλιογραφία επιδεικνύει μια ποικιλία προσεγγίσεων που χρησιμοποιούν το Web Audio API, προκειμένου να ολοκληρώσει τον ήχο σε web περιβάλλον [44], [45], [46], [47]. Επιπλέον, πολλοί ερευνητές έχουν προτείνει διάφορες μεθόδους ρύθμισης του Web Audio API στην εφαρμογή τους [48], [49], [50].

Σε κάθε περίπτωση, η προσέγγιση του το Web Audio API βασίζεται στην έννοια του πλαισίου του ήχου, το οποίο παρουσιάζει την κατεύθυνση των ρευμάτων των ηχητικών ροών, μεταξύ των ηχητικών κόμβων. Σε κάθε κόμβο οι ιδιότητες του ήχου μπορούν να προσαρμοστούν και να αλλάξουν, ανάλογα με τις απαιτήσεις της εφαρμογής. Οι κόμβοι που περιλαμβάνονται είναι δυνατόν να ταξινομηθούν με βάση τον τύπο: α) κόμβοι πηγής (audio buffers, live audio inputs, oscillators και JS processors), β) κόμβοι τροποποίησης (filters, convolvers και panners), γ) κόμβοι ανάλυσης (analyzers και JS processors), δ) κόμβοι προορισμού (audio outputs και offline processing buffers) [29].

## 2.2.5 Audio

### 2.2.5.1 3D Audio

Ο ήχος αποτελεί ένα θεμελιώδες μέρος των εφαρμογών στην καθημερινότητα και μπορεί να παραχθεί από ένα πλήθος ηχητικών πηγών. Κάθε ήχος έχει μια συγκεκριμένη κατεύθυνση και μπορεί να αναγνωριστεί εύκολα, λόγω των διακριτικών χαρακτηρισμών του και την οικειότητα που έχουμε αναπτύξει με τους ήχους [24]. Επιπλέον, ένα χαρακτηριστικό του συστήματος της ανθρώπινης ακοής είναι η ικανότητα να αντιλαμβάνεται τον 3D ήχο.

Γενικά, η έννοια του 3D ήχου εστιάζεται στον τρόπο που ο ακροατής αντιλαμβάνεται τους εισερχόμενους ήχους από όλες τις κατευθύνσεις. Η έννοια του 3D ήχου είναι το ίδιο όταν προσομοιώνεται από έναν υπολογιστή [25]. Με άλλα λόγια, ένας ακροατής μπορεί να αναγνωρίσει ουσιαστικό σημείο αναφοράς στο χώρο από μια πηγή ήχου, για παράδειγμα την κατεύθυνση, την απόσταση και την ευρυχωρία [26]. Σύμφωνα με τα παραπάνω, ο 3D ήχος είναι σημαντικό να συμπεριληφθεί σε εικονικούς κόσμους, προκειμένου να βελτιωθεί ο ρεαλισμός και καθηλωτική αίσθηση στη σκηνή.

Σε αυτή την περίπτωση, ο χρήστης μπορεί να καταλάβει οποιαδήποτε πηγή ήχου η οποία περιλαμβάνεται στην εικονική 3D σκηνή μέσω του συνδυασμού οπτικής και ακουστικής αίσθησης.

Με την προσθήκη φυσικού χωρικού ήχου σε διαδραστικές 3D εφαρμογές, μεταφέρεται μια πιο εξελιγμένη πληροφορία, σε συνδυασμό με τη χρήση άλλων μέσων (π.χ., όραση) [27].

### 2.2.5.2 Web 3D Audio

Πολλές προσεγγίσεις για τις Web 3D εφαρμογές αποσκοπούν στη ρεαλιστική απεικόνιση της σκηνής. Πολύ συχνά παραβλέπεται το σημείο του ήχου, αν και μπορεί να προσφέρει περισσότερες λεπτομέρειες στα 3D γραφικά. Συγκεκριμένα, ένα υψηλό επίπεδο εμπάθυσης επιτυγχάνεται και η φυσική αλληλεπίδραση αυξάνεται, λόγω του ότι η γραφική σκηνή προσομοιάζει τον πραγματικό κόσμο με τον καλύτερο δυνατό τρόπο [23]. Εξίσου σημαντική είναι η χωρική ακουστική και η απόδοση των χωρικών χαρακτηριστικών του κάθε ακουστικού αντικειμένου, σε ένα Web 3D κόσμο. Αυτά τα χαρακτηριστικά περιλαμβάνουν αντιληπτές οδηγίες, αποστάσεις και χωρικές επεκτάσεις των ακουστικών αντικειμένων. Επιπλέον, αυτά τα χαρακτηριστικά θα πρέπει να κατανοηθούν με τον ίδιο τρόπο όπως αυτά αναγνωρίζονται στον πραγματικό κόσμο [28].

Για όλους τους παραπάνω λόγους, έχει δοθεί ιδιαίτερη προσοχή για την εισαγωγή του ήχου σε διαδικτυακή εφαρμογή. Η πρώτη προσπάθεια έγινε μέσω της ετικέτας <bgsound>, στην οποία μόνο μουσική υπόκρουση θα μπορούσε να περιέχεται σε μια ιστοσελίδα και ήταν διαθέσιμη για συγκεκριμένα προγράμματα περιήγησης. Μετά από αυτό, το Flash ήταν ο πρώτος τρόπος cross-browser του ήχου στο Web, αλλά απαιτήθηκαν plugins. Το επίκεντρο της πρόσφατης έρευνας επικεντρώθηκε στο στοιχείο <audio> σε HTML5, το οποίο θα μπορούσε να αποφύγει τα plugins, αλλά δεν προοριζόταν για περίπλοκες και φιλόδοξες υλοποιήσεις [29], [30]. Συγκεκριμένα, το στοιχείο <audio> είναι κατώτερο στην εφαρμογή φίλτρων στο σήμα του ήχου και στην πρόσβαση των ακατέργαστων PCM δεδομένων. Επιπλέον, δεν περιλαμβάνει την έννοια της θέσης και της κατεύθυνσης των πηγών και των ακροατών.

Τέλος, δεν παρέχει μοντέλο ακριβούς χρόνου σε χαμηλή καθυστέρηση, το οποίο είναι πολύ σημαντικό στις διαδραστικές εφαρμογές για την ανάγκη της γρήγορης ακουστικής απόκρισης στις ενέργειες του χρήστη [31]. Έτσι, δεν είναι επαρκές για ένα 3D διαδραστικό δικτυακό περιβάλλον με απαιτητικό ηχητικό σχεδιασμό. Υπό αυτές τις συνθήκες, πολλές εναλλακτικές λύσεις έχουν προταθεί, προκειμένου να καθιερωθεί ένα αποτελεσματικό API, το οποίο θα φροντίζει να ξεπεραστούν τα περισσότερα από αυτά τα όρια. Μία πιο ενδιαφέρουσα προσέγγιση στο θέμα αυτό είναι το Web Audio API, το οποίο έχει προταθεί από το Mozilla Foundation. Προκάτοχός του Web Audio API ήταν το Audio Data API. Αυτό το API παρείχε μια ξεχωριστή δομή για την εγγραφή ηχητικών ανακλήσεων σε JavaScript, αλλά δεν παρέχει προδιαγραφές για χαμηλού επιπέδου, έμφυτο, προ-μεταγλωττισμένο μέσο που πρέπει να περιλαμβάνονται στα προγράμματα περιήγησης [32]. Ως αποτέλεσμα, το Mozilla Foundation κατέληξε στην έκδοση του Web Audio API, το οποίο εγκρίθηκε από τα άλλα προγράμματα περιήγησης.

## 3 Σχέδιο δράσης για την εκπόνηση της εργασίας

### 3.1 State of the Art

Το κεφάλαιο αυτό ασχολείται με την βιβλιογραφική αναζήτηση της τεχνολογίας αιχμής (state of the art) και την υλοποίηση του σχεδίου δράσης για την εκπόνηση της πτυχιακής εργασίας.

#### 3.1.1 X3D

##### Εισαγωγή

Το X3D (eXtensible 3D) είναι μορφή αρχείων ανοικτών προτύπων, χωρίς δικαιώματα βασισμένα σε XML για την αναπαράσταση τρισδιάστατων γραφικών. Το X3D προορίζεται για χρήση σε μια ποικιλία από hardware και σε διάφορους τομείς εφαρμογών, όπως η μηχανική και επιστημονική απεικόνιση, σε παρουσιάσεις πολυμέσων, στην ψυχαγωγία, τις ιστοσελίδες, και σε εικονικούς κόσμους. Για τους σκοπούς αυτής της πτυχιακής θα αναφέρουμε μόνο τη βασική δομή του ήχου ενός X3D αρχείου.

Σε αυτή την ενότητα θα γίνει περιγραφή του στοιχείου του ήχου (sound component). Επίσης, περιλαμβάνει το πώς ο ήχος παραδίδεται σε έναν κόσμο X3D, καθώς και τον τρόπο πρόσβασης στους ήχους.

##### Ήχος

Στην ενότητα αυτή θα γίνει μια ανάλυση των πεδίων (fields) του κόμβου του ήχου (Sound node) του X3D. Στην παρακάτω εικόνα παρουσιάζεται η δομή του.

```
Sound : X3DSoundNode {
  SFVec3f [in,out] direction 0 0 1 (-∞,∞)
  SFFloat [in,out] intensity 1 [0,1]
  SFVec3f [in,out] location 0 0 0 (-∞,∞)
  SFFloat [in,out] maxBack 10 [0,∞)
  SFFloat [in,out] maxFront 10 [0,∞)
  SFNode [in,out] metadata NULL [X3DMetadataObject]
  SFFloat [in,out] minBack 1 [0,∞)
  SFFloat [in,out] minFront 1 [0,∞)
  SFFloat [in,out] priority 0 [0,1]
  SFNode [in,out] source NULL [X3DSoundSourceNode]
  SFBool [] spatialize TRUE
}
```

Εικόνα 1- Δομή του κόμβου του ήχου



Ο κόμβος του ήχου καθορίζει τη χωρική παρουσίαση ενός ήχου σε μια σκηνή Χ3D. Ο ήχος βρίσκεται σε ένα σημείο στο τοπικό σύστημα συντεταγμένων και εκπέμπει ήχο σε μια ελλειπτική τροχιά που ορίζεται από δύο ελλειψοειδή. Τα ελλειψοειδή προσανατολίζονται σε μια κατεύθυνση που καθορίζεται από το πεδίο της κατεύθυνσης. Το σχήμα των ελλειψοειδών μπορεί να τροποποιηθεί για να παρέχει περισσότερη ή λιγότερη κατευθυντική εστίαση από τη θέση του ήχου.

Το πεδίο της πηγής (source) προσδιορίζει την πηγή του ήχου για τον κόμβο του ήχου. Αν το πεδίο της πηγής δεν έχει καθοριστεί, ο κόμβος του ήχου δεν θα εκπέμπει ήχο. Το πεδίο της πηγής προσδιορίζει είτε κόμβο AudioClip ή έναν κόμβο MovieTexture. Αν καθοριστεί ένας κόμβος MovieTexture ως πηγή ήχου, το MovieTexture παραπέμπει σε μια μορφή ταινίας που υποστηρίζει ήχο.

Το πεδίο της έντασης (intensity) ρυθμίζει την ένταση (ντεσιμπέλ) του ήχου που εκπέμπεται από τον κόμβο του ήχου. Το πεδίο της έντασης έχει μια τιμή που κυμαίνεται 0.0 έως 1.0 και καθορίζει ένα παράγοντα που θα πρέπει να χρησιμοποιείται για τη βαθμονόμηση του ομαλοποιημένου δείγματος δεδομένων της πηγής του ήχου κατά την αναπαραγωγή. Ένας κόμβος ήχου με ένταση 1.0 εκπέμπει ήχο με τη μέγιστη ένταση (πριν την εξασθένηση), και ένας κόμβος ήχου με ένταση 0.0 δεν πρέπει να εκπέμπει ήχο. Μεταξύ αυτών των τιμών, η ένταση θα πρέπει να αυξηθεί γραμμικά από -20 dB σε 0 dB πλησιάζοντας την ένταση από 0.0 σε 1.0.

Το πεδίο της προτεραιότητας (priority) παρέχει ένα στοιχείο για το πρόγραμμα περιήγησης για να επιλέξει ποιός ήχος θα αναπαραχθεί, όταν υπάρχουν περισσότεροι ενεργοί κόμβοι ήχου από όσους μπορούν να αναπαραχθούν ταυτόχρονα είτε λόγω των περιορισμένων πόρων του συστήματος ή λόγω του φορτίου του συστήματος. Το πεδίο προτεραιότητας κυμαίνεται 0.0 έως 1.0, με 1.0 είναι η υψηλότερη προτεραιότητα και 0.0 η χαμηλότερη προτεραιότητα.

Το πεδίο της τοποθεσίας (location) καθορίζει τη θέση του ηχητικού πομπού στο τοπικό σύστημα συντεταγμένων. Η έξοδος ενός ηχητικού κόμβου ακούγεται μόνο αν είναι μέρος της διασχίζουσας σκηνής. Οι ηχητικοί κόμβοι που κατάγονται από LOD, Switch, ή οποιαδήποτε ομάδα ή πρωτότυπο κόμβο που απενεργοποιεί διάσχιση των παιδιών του δεν ακούγεται, εκτός εάν διασχίζεται. Αν ένας ηχητικός κόμβος έχει απενεργοποιηθεί από ένα Switch ή LOD κόμβο και αργότερα γίνει μέρος της διάσχισης και πάλι, ο ήχος θα συνεχιστεί από το σημείο αυτό που θα ήταν εάν έπαιζε συνεχόμενα.

Ο ηχητικός κόμβος έχει ένα εσωτερικό ελλειψοειδή που ορίζει ένα όγκο του χώρου στον οποίο η μέγιστη στάθμη του ήχου μπορεί να ακουστεί. Μέσα σε αυτό το ελλειψοειδή, το ομαλοποιημένο δείγμα δεδομένων κλιμακώνεται από το πεδίο της έντασης και δεν υπάρχει εξασθένηση. Το εσωτερικό ελλειψοειδή ορίζεται από την επέκταση του διανύσματος κατεύθυνσης μέσω της τοποθεσίας. Τα πεδία minBack και minFront καθορίζουν τις αποστάσεις πίσω και μπροστά από τη θέση κατά μήκος του διανύσματος κατεύθυνσης αντίστοιχα. Το εσωτερικό ελλειψοειδή έχει μία από τις εστίες του στη θέση και τέμνει το διάνυσμα κατεύθυνσης στο minBack και το minFront.

Ο ηχητικός κόμβος έχει και ένα εξωτερικό ελλειψοειδή που ορίζει ένα όγκο του χώρου που περιβάλλει την ακουστικότητα του ήχου. Δεν μπορεί να ακουστεί ήχος έξω από αυτό το εξωτερικό ελλειψοειδή. Το εξωτερικό ελλειψοειδή ορίζεται από την επέκταση του διανύσματος κατεύθυνσης μέσω της τοποθεσίας. Τα πεδία maxBack και maxFront καθορίζουν τις αποστάσεις πίσω και μπροστά από τη θέση κατά μήκος του διανύσματος κατεύθυνσης αντίστοιχα. Το εξωτερικό ελλειψοειδή έχει μία από τις εστίες του στη θέση και τέμνει το διάνυσμα κατεύθυνσης στο maxBack και το maxFront.

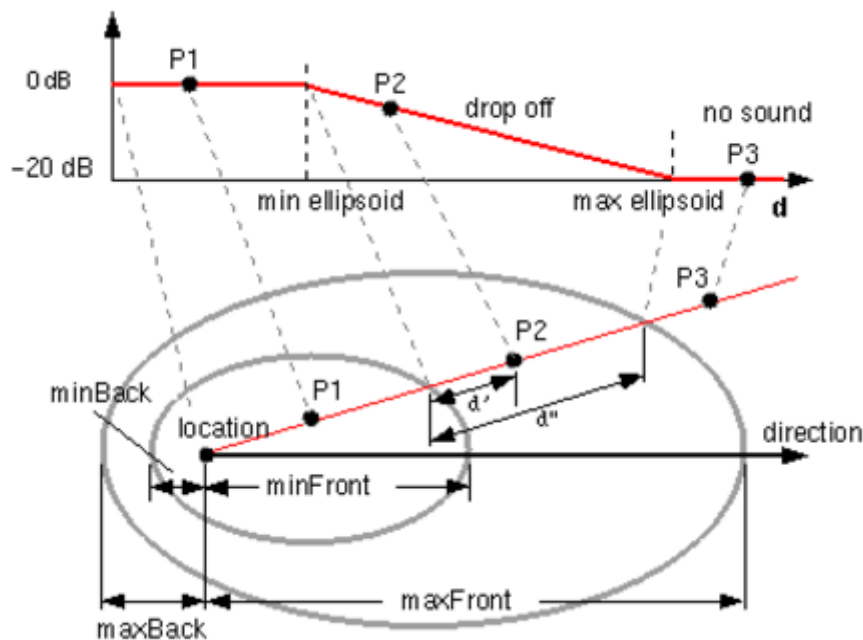
Τα πεδία minFront, maxFront, minBack και maxBack ορίζονται στις τοπικές συντεταγμένες, και πρέπει να είναι μεγαλύτερα ή ίσα με το μηδέν. Το πεδίο minBack πρέπει να είναι μικρότερο ή

ίσο με το  $\max\text{Back}$ , και το  $\min\text{Front}$  πρέπει να είναι μικρότερο ή ίσο με το  $\max\text{Front}$ . Οι παράμετροι του ελλειψοειδούς ορίζονται στο τοπικό σύστημα συντεταγμένων, αλλά η γεωμετρία των ελλειψοειδών επηρεάζεται από τη μετατροπή των προγόνων τους.

Μεταξύ των δύο ελλειψοειδών, πρέπει να υπάρχει μια γραμμική βαθμίδα εξασθένησης στην ένταση του ήχου, από 0 dB στο ελάχιστο ελλειψοειδή έως -20 dB στη μέγιστο ελλειψοειδή. Παρακάτω φαίνεται ο τύπος για τον υπολογισμό αυτής της εξασθένησης.

$$\text{attenuation} = -20 \times (d' / d'')$$

όπου το  $d$  είναι η απόσταση κατά μήκος του διανύσματος από την θέση στο θεατή, και μετράται από τη μετατροπή του ελάχιστου ορίου του ελλειψοειδούς στον θεατή, και το  $d''$  είναι η απόσταση κατά μήκος του διανύσματος από την θέση στο θεατή από τη μετατροπή του ελάχιστου ορίου του ελλειψοειδούς στη μετατροπή του μέγιστου ορίου του ελλειψοειδούς.



Εικόνα 2- Γεωμετρία ηχητικού κόμβου

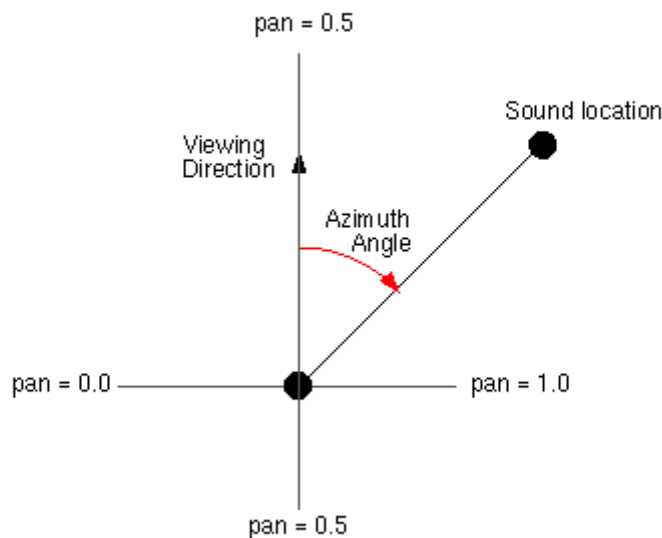
### Ηχητική εξασθένηση και χωροθέτηση

Προκειμένου να δημιουργηθεί μια γραμμική μείωση στην ένταση του ήχου καθώς ο θεατής κινείται από το εσωτερικό προς την εξωτερική ελλειψοειδή του ήχου, η εξασθένηση πρέπει να βασίζεται σε μια γραμμική βαθμίδα ντεσιμπέλ. Για να γίνει μια συνεπή πτώση σε όλους τους περιηγητές, η βαθμίδα ντεσιμπέλ κυμαίνεται από 0 dB στο ελάχιστο του ελλειψοειδούς έως -20 dB στο εξωτερικό του ελλειψοειδούς. Οι ηχητικοί κόμβοι με ένα εξωτερικό ελλειψοειδή που είναι δέκα φορές μεγαλύτερο από το ελάχιστο, θα εμφανιστεί το αντίστροφο τετράγωνο μείωσης της έντασης που προσεγγίζει εξασθένηση του ήχου σε περιβάλλον χωρίς ηχώ.

Τα προγράμματα περιήγησης μπορούν να υποστηρίξουν χωρικό περιορισμό των ήχων στους οποίους το πεδίο  $\text{spatialize}$  έχει την τιμή "TRUE" καθώς το επιτρέπουν οι βασικές βιβλιοθήκες ήχου.

Τα προγράμματα περιήγησης πρέπει τουλάχιστον υποστηρίζουν στερεοφωνικό panning για ήχους που δεν είναι MIDI με βάση τη γωνία μεταξύ του θεατή και της πηγής. Αυτή η γωνία επιτυγχάνεται με την προβολή της θέσης του ήχου πάνω στο επίπεδο XZ του θεατή. Ο καθορισμός της γωνίας μεταξύ του άξονα Z και του διανύσματος από τον θεατή στην μεταφερόμενη θέση του, ορίζοντας την τιμή του pan στο εύρος [0.0, 1.0], όπως απεικονίζεται στο παρακάτω σχήμα. Με δεδομένη αυτή τη τιμή του pan, το επίπεδο του αριστερού και δεξιού καναλιού μπορεί να υπολογιστεί χρησιμοποιώντας τις ακόλουθες εξισώσεις:

$$\begin{aligned} \text{leftPanFactor} &= 1 - \text{pan}^2 \\ \text{rightPanFactor} &= 1 - (1 - \text{pan})^2 \end{aligned}$$



Εικόνα 3- Στερεοφωνικό panning

Χρησιμοποιώντας αυτή την τεχνική, η ένταση του ήχου τροποποιείται από την τιμή του πεδίου της έντασης, και ύστερα η απόσταση εξασθένησης για να υπολογιστεί η μη-χωρική έξοδος του ήχου. Οι τιμές στην μη-χωρική έξοδο του ήχου είναι η κλίμακα από το *leftPanFactor* και το *rightPanFactor* που καθορίζουν τα τελικά σήματα εξόδου στο αριστερό και το δεξί κανάλι.

### Κόμβος AudioClip

Στην ενότητα αυτή θα γίνει μια ανάλυση των πεδίων (fields) του κόμβου AudioClip του X3D. Ένας κόμβος AudioClip καθορίζει τα δεδομένα του ήχου που μπορεί να αναφερθούν από τον ηχητικό κόμβο. Στην παρακάτω εικόνα παρουσιάζεται η δομή του.

```

AudioClip : X3DSoundSourceNode, X3DUrlObject {
  SFString [in,out] description      ""
  SFBool   [in,out] loop             FALSE
  SFNode   [in,out] metadata        NULL [X3DMetadataObject]
  SFTime   [in,out] pauseTime       0    (-∞,∞)
  SFFloat  [in,out] pitch           1.0  (0,∞)
  SFTime   [in,out] resumeTime      0    (-∞,∞)
  SFTime   [in,out] startTime       0    (-∞,∞)
  SFTime   [in,out] stopTime        0    (-∞,∞)
  MFString [in,out] url              []   [URI]
  SFTime   [out]   duration_changed
  SFTime   [out]   elapsedTime
  SFBool   [out]   isActive
  SFBool   [out]   isPaused
}

```

Εικόνα 4- Δομή του κόμβου AudioClip

Το πεδίο της περιγραφής (description) καθορίζει μια κειμενική περιγραφή της πηγής ήχου. Ένα πρόγραμμα περιήγησης δεν απαιτείται να εμφανίσει το πεδίο της περιγραφής, αλλά μπορεί να επιλέξουν να το εμφανίσουν για την αναπαραγωγή του ήχου.

Το πεδίο url καθορίζει τη διεύθυνση URL από την οποία φορτώνεται ο ήχος. Τα προγράμματα περιήγησης υποστηρίζουν τουλάχιστον τη μορφή wavfile σε μη συμπιεσμένη μορφή PCM. Συνιστάται για τα προγράμματα περιήγησης να υποστηρίζουν επίσης τον τύπο 1 της μορφής MIDI και τη συμπιεσμένη μορφή MP3. Τα αρχεία MIDI θεωρούνται ότι χρησιμοποιούν General MIDI patch.

Το πεδίο pitch καθορίζει ένα πολλαπλασιαστή για το ρυθμό με τον οποίο αναπαράγεται το δείγμα ήχου. Οι τιμές για το πεδίο pitch πρέπει να είναι μεγαλύτερες από το μηδέν. Η αλλαγή του πεδίου pitch επηρεάζει τόσο το pitch αλλά και την ταχύτητα αναπαραγωγής του ήχου. Ένα γεγονός set\_pitch σε ένα ενεργό AudioClip αγνοείται και δεν δημιουργείται πεδίο pitch\_changed. Εάν το pitch έχει οριστεί σε 2.0, ο ήχος θα πρέπει να αναπαράγεται σε μια οκτάβα υψηλότερα από το κανονικό και δύο φορές πιο γρήγορα. Για ένα δείγμα ήχου το πεδίο pitch μεταβάλλει το ρυθμό δειγματοληψίας στο οποίο αναπαράγεται ο ήχος. Η σωστή εφαρμογή του ελέγχου του pitch για MIDI είναι ο πολλαπλασιασμός του ρυθμού της αναπαραγωγής με την τιμή του pitch και η ρύθμιση του ελέγχου MIDI Coarse Tune και Fine Tune για να επιτευχθεί η σωστή αλλαγή στο pitch.

Ένα γεγονός duration\_changed αποστέλλεται κάθε φορά που υπάρχει μια νέα τιμή για την "κανονική" διάρκεια του κλιπ. Συνήθως, αυτό θα συμβεί μόνο όταν η τρέχουσα διεύθυνση URL που χρησιμοποιείται αλλάζει και τα δεδομένα του ήχου έχουν φορτωθεί, υποδεικνύοντας ότι το κλιπ αναπαράγει διαφορετική ηχητική πηγή. Η διάρκεια είναι το μήκος του χρόνου σε δευτερόλεπτα για έναν κύκλο του ήχου όταν η τιμή του pitch είναι 1.0. Η αλλαγή του πεδίου pitch δεν θα προκαλέσει ένα γεγονός duration\_changed. Η τιμή "-1" του duration σημαίνει ότι τα δεδομένα ήχου δεν έχουν ακόμη φορτωθεί ή η τιμή δεν είναι διαθέσιμη για κάποιο λόγο. Ένα γεγονός duration\_changed πρέπει να δημιουργηθεί αν ο κόμβος AudioClip φορτώνεται όταν διαβάζεται το αρχείο X3D ή όταν ο κόμβος AudioClip προστίθεται στο γράφημα σκηνής.

Το πεδίο isActive μπορεί να χρησιμοποιηθεί από άλλους κόμβους για να καθορίσει εάν το κλιπ είναι ενεργό. Αν ένα AudioClip είναι ενεργό, πρέπει η αναπαραγωγή του ήχου να αντιστοιχεί στον χρόνο του ήχου (δηλαδή, σε τοπικό σύστημα ώρας του ήχου με το δείγμα 0 τη χρονική στιγμή 0):

$$t = (\text{now} - \text{startTime}) \text{ modulo } (\text{duration} / \text{pitch})$$

Τα πεδία εισόδου-εξόδου `loop`, `pauseTime`, `resumeTime`, `startTime`, και `stopTime` περιλαμβάνουν τιμές για την συνεχή αναπαραγωγή, τον χρόνο παύσης, τον χρόνο συνέχειας στην αναπαραγωγή μετά από παύση, τον χρόνο έναρξης και τον χρόνο του τέλους της αναπαραγωγής ενός ήχου, ενώ τα πεδία μόνο εξόδου `elapsedTime`, `isActive`, και `isPaused` περιλαμβάνουν τιμές για τον χρόνο/ διάρκεια που βρίσκεται σε αναπαραγωγή ο ήχος, αν είναι ενεργός και αν βρίσκεται σε παύση.

### 3.1.2 Web Audio API

#### Εισαγωγή

Ο ήχος στο διαδίκτυο βρίσκεται μέχρι σήμερα σε πρωτόγονο στάδιο και μέχρι πολύ πρόσφατα έπρεπε να παρέχεται μέσω plugins όπως το Flash και QuickTime. Η εισαγωγή του στοιχείου του ήχου στο HTML5 είναι πολύ σημαντική, γιατί επιτρέπει την βασική αναπαραγωγή ήχου συνεχούς ροής. Αλλά δεν είναι αρκετά ισχυρή για να χειριστεί πιο σύνθετες εφαρμογές ήχου. Για εξελιγμένα web-based παιχνίδια ή διαδραστικές εφαρμογές, είναι απαραίτητη μια άλλη λύση. Ένας στόχος αυτής της προδιαγραφής είναι να συμπεριλάβει τις δυνατότητες που βρέθηκαν σε σύγχρονες μηχανές ήχου παιχνιδιών, καθώς και ορισμένες από τις λειτουργίες μίξης, επεξεργασίας και φιλτραρίσματος που βρίσκονται σε σύγχρονες εφαρμογές παραγωγής ήχου.

Τα APIs έχουν σχεδιαστεί με μια ευρεία ποικιλία σεναρίων χρήσης [webaudio-usecases]. Στην ιδανική περίπτωση, θα πρέπει να είναι σε θέση να υποστηρίξει κάθε περίπτωση χρήσης η οποία θα μπορούσε λογικά να εφαρμοστεί με μια βελτιστοποιημένη C++ μηχανή ελεγχόμενη μέσω JavaScript και να εκτελεστεί σε ένα πρόγραμμα περιήγησης. Όπως αναφέρθηκε, ένα σύγχρονο λογισμικό desktop ήχου μπορεί να έχει πολύ προηγμένες δυνατότητες, μερικές από τις οποίες θα ήταν δύσκολο ή αδύνατο να χτίσουμε με αυτό το σύστημα. Η Apple Logic είναι μια εφαρμογή η οποία έχει την υποστήριξη εξωτερικών ελεγκτών MIDI, αυθαίρετα plugin ηχητικών εφέ και συνθεσάιζερ, ιδιαίτερα βελτιστοποιημένο direct-to-disk αρχείο ήχου ανάγνωσης/ γραφής, ενσωματωμένο time-stretching, και ούτω καθεξής.

Παρ' όλα αυτά, το προτεινόμενο σύστημα θα είναι σε θέση να υποστηρίξει ένα μεγάλο εύρος λογικών περίπλοκων παιχνιδιών και διαδραστικών εφαρμογών, συμπεριλαμβανομένων και των μουσικών. Και αυτό μπορεί να είναι ένα πολύ καλό συμπλήρωμα για τα πιο προηγμένα γραφικά χαρακτηριστικά που προσφέρει το WebGL. Το API έχει σχεδιαστεί έτσι ώστε να μπορούν να προστεθούν πιο προηγμένες δυνατότητες σε μεταγενέστερο στάδιο.

#### 3.1.2.1 Χωρικοποίηση (Spatialization/Panning)

#### Υπόβαθρο

Ένα κοινό χαρακτηριστικό των απαιτήσεων των σύγχρονων 3D παιχνιδιών είναι η ικανότητα του δυναμικού χωρικού ήχου και η μετακίνηση πολλαπλών πηγών ήχου σε 3D χώρο. Οι μηχανές

game audio, όπως η OpenAL, η FMOD, η Creative EAX, η XACT Audio της Microsoft, κλπ έχουν αυτή την ικανότητα.

Χρησιμοποιώντας το PannerNode, μια ροή ήχου μπορεί να χωρικοποιηθεί ή να τοποθετηθεί στο χώρο σε σχέση με τον AudioListener. Ένα AudioContext θα περιλαμβάνει έναν μόνο AudioListener. Και οι δύο, panners και listeners, έχουν μια θέση στον 3D χώρο σύμφωνα με το καρτεσιανό σύστημα συντεταγμένων χρησιμοποιώντας τον κανόνα του δεξιού χεριού. Οι μονάδες που χρησιμοποιούνται στο σύστημα συντεταγμένων δεν ορίζονται, και δεν χρειάζεται να οριστούν επειδή τα αποτελέσματα που υπολογίζονται με τις συντεταγμένες αυτές είναι ανεξάρτητα/αμετάβλητα οποιωνδήποτε συγκεκριμένων μονάδων όπως τα μέτρα ή τα πόδια. Τα αντικείμενα PannerNode (που αντιπροσωπεύουν τη ροή πηγής) έχουν ένα διάνυσμα προσανατολισμού που εκπροσωπούν σε ποια κατεύθυνση προβάλλεται ο ήχος. Επιπλέον, έχουν ένα κωνικό ήχο που αντιπροσωπεύει το πόσο κατευθυντήριος είναι ο ήχος.

Για παράδειγμα, ο ήχος θα μπορούσε να είναι πολυκατευθυντικός, όπου στη συγκεκριμένη περίπτωση θα πρέπει να ακούγεται παντού, ανεξάρτητα από τον προσανατολισμό του, ή μπορεί να είναι πιο κατευθυντήριος και να ακούγεται μόνο αν είναι στραμμένος προς τον ακροατή. Τα αντικείμενα του AudioListener (που αντιπροσωπεύουν τα αυτιά ενός ατόμου) έχουν προσανατολισμό και διάνυσμα που αντιπροσωπεύει την κατεύθυνση στην οποία κοιτάει το άτομο. Επειδή η ροή πηγής και ο ακροατής μπορεί να κινούνται, και οι δύο έχουν ένα διάνυσμα ταχύτητας που αντιπροσωπεύει τόσο την ταχύτητα αλλά και την κατεύθυνση της κίνησης. Από κοινού, αυτές οι δύο ταχύτητες μπορούν να χρησιμοποιηθούν για τη δημιουργία ενός εφέ Doppler το οποίο αλλάζει τον τόνο του ήχου.

Κατά τη διάρκεια του rendering, το PannerNode υπολογίζει ένα αζιμούθιο και την ανύψωση. Οι τιμές αυτές χρησιμοποιούνται εσωτερικά από την εφαρμογή, προκειμένου να δημιουργήσει το εφέ της χωροθέτησης (spatialization).

### **Αζιμούθιο και Ανύψωση (Azimuth and Elevation)**

Ο παρακάτω αλγόριθμος χρησιμοποιείται για τον υπολογισμό του αζιμούθιου και της ανύψωση για το PannerNode:

```
// Υπολογισμός του διανύσματος πηγής-ακροατή.  
vec3 sourceListener = source.position - listener.position;  
  
if (sourceListener.isZero()) {  
    // Υπολογισμός σε περίπτωση που η πηγή και ο ακροατής είναι στο ίδιο σημείο.  
    azimuth = 0;  
    elevation = 0;  
    return;  
}  
  
sourceListener.normalize();  
  
// Ευθυγράμμιση αξόνων.  
vec3 listenerFront = listener.orientation;  
vec3 listenerUp = listener.up;
```

```
vec3 listenerRight = listenerFront.cross(listenerUp);
listenerRight.normalize();

vec3 listenerFrontNorm = listenerFront;
listenerFrontNorm.normalize();

vec3 up = listenerRight.cross(listenerFrontNorm);

float upProjection = sourceListener.dot(up);

vec3 projectedSource = sourceListener - upProjection * up;
projectedSource.normalize();

azimuth = 180 * acos(projectedSource.dot(listenerRight)) / PI;

// Πηγή μπροστά ή πίσω από τον ακροατή.
float frontBack = projectedSource.dot(listenerFrontNorm);
if (frontBack < 0)
    azimuth = 360 - azimuth;

// Κάνοντας το αζιμούθιο σχετικό με το "μπροστινό" και όχι "δεξιό" διάνυσμα του ακροατή.
if ((azimuth >= 0) && (azimuth <= 270))
    azimuth = 90 - azimuth;
else
    azimuth = 450 - azimuth;

elevation = 90 - 180 * acos(sourceListener.dot(up)) / PI;

if (elevation > 90)
    elevation = 180 - elevation;
else if (elevation < -90)
    elevation = -180 - elevation;
```

### Αλγόριθμος Panning

Πρέπει να υποστηρίζεται panning από μονοφωνική σε στερεοφωνική και από στερεοφωνική σε στερεοφωνική. Η επεξεργασία από μονοφωνική σε στερεοφωνική χρησιμοποιείται όταν όλες οι συνδέσεις στην είσοδο είναι μονοφωνικές. Διαφορετικά χρησιμοποιείται επεξεργασία από στερεοφωνική σε στερεοφωνική.

### Panning ίσης ισχύος (Equal-power panning)

Αυτός είναι ένας απλός και σχετικά οικονομικός αλγόριθμος που περιλαμβάνει βασικά, αλλά λογικά αποτελέσματα. Χρησιμοποιείται για το StereoPannerNode, και για το PannerNode όταν η

τιμή του μοντέλου ranning έχει οριστεί σε "equalpower", όπου στην περίπτωση αυτή αγνοείται η τιμή της ανύψωσης. Αυτός ο αλγόριθμος πρέπει να υλοποιηθεί με τη χρήση a-rate παραμέτρων, που σημαίνει ότι πρέπει να ληφθούν υπόψη οι τιμές σε μια βάση ανά δείγμα ήχου.

1. Για κάθε δείγμα που θα υπολογιστεί από το AudioNode:

1. Το αζιμούθιο υπολογίζεται σύμφωνα με την ενότητα **Αζιμούθιο και Ανύψωση**.
2. Η τιμή του αζιμούθιου περιορίζεται πρώτα ανάμεσα στο εύρος [-90, 90] σύμφωνα με τα παρακάτω:

```
// Πρώτα, ενσωματώνουμε το αζιμούθιο στο επιτρεπτό όριο εύρους [-180, 180].
azimuth = max(-180, azimuth);
azimuth = min(180, azimuth);
```

```
// Μετά το προσαρμόζουμε ανάμεσα στο εύρος [-90, 90].
```

```
if (azimuth < -90)
    azimuth = -180 - azimuth;
else if (azimuth > 90)
    azimuth = 180 - azimuth;
```

3. Μια ομαλοποιημένη τιμή x υπολογίζεται από το αζιμούθιο για μία μονοφωνική είσοδο ως εξής:

```
x = (azimuth + 90) / 180;
```

Και για μία στερεοφωνική είσοδο ως εξής:

```
if (azimuth <= 0) { // Αν το αζιμούθιο κυμαίνεται από -90 ~ 0 μοίρες
// Μεταφέρουμε την τιμή του αζιμούθιου από το εύρος [-90, 0] μοιρών στο εύρος [-90, 90].
x = (azimuth + 90) / 90;
} else { // Αλλιώς αν το αζιμούθιο κυμαίνεται από 0 ~ 90 μοίρες
// Μεταφέρουμε την τιμή του αζιμούθιου από το εύρος [0, 90 μοιρών στο εύρος [-90, 90].
x = azimuth / 90;
}
```

Για το StereoPannerNode, πρέπει να εφαρμοστεί ο ακόλουθος αλγόριθμος.

1. Για κάθε δείγμα που θα υπολογιστεί από το AudioNode:

1. Το pan υπολογίζεται σύμφωνα με την μεταβλητή pan του AudioParam της ενότητας **Διεπαφή StereoPannerNode**.
2. Περιορίζουμε το pan στο εύρος [-1, 1].



```
pan = max(-1, pan);
pan = min(1, pan);
```

- Υπολογίζουμε το  $x$  ομαλοποιώντας την τιμή του  $pan$  στο εύρος  $[0, 1]$  για τη μονοφωνική είσοδο ως εξής:  
 $x = (pan + 1) / 2;$

Και για τη στερεοφωνική είσοδο ως εξής:

```
if (pan <= 0)
    x = pan + 1;
else
    x = pan;
```

Τα επόμενα βήματα ακολουθούνται για να επιτύχουμε panning ίσης ισχύος:

- Οι τιμές του  $gain$  για αριστερά και δεξιά υπολογίζονται ως εξής:

```
gainL = cos(x * Math.PI / 2);
gainR = sin(x * Math.PI / 2);
```

- Για μονοφωνική είσοδο, η στερεοφωνική έξοδος υπολογίζεται ως εξής:

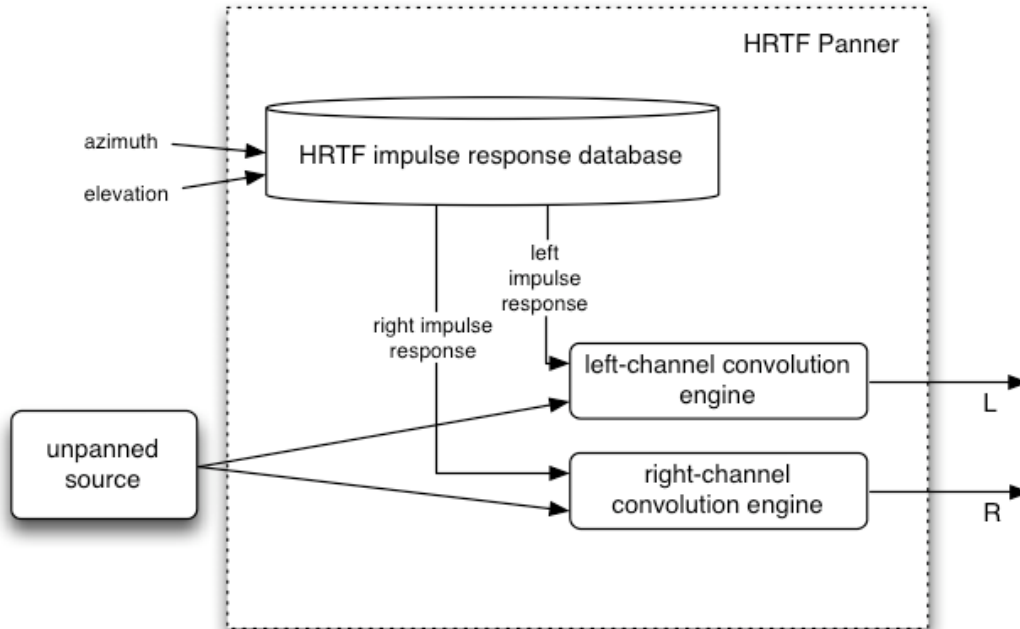
```
outputL = input * gainL;
outputR = input * gainR;
```

Για στερεοφωνική είσοδο, η έξοδος υπολογίζεται ως εξής:

```
if (pan <= 0) {
// Το πέρασμα από το inputL στο outputL και με pan ίσης ισχύος στο inputR όπως σε μονοφωνική
περίπτωση.
    outputL = inputL + inputR * gainL;
    outputR = inputR * gainR;
} else {
// Το πέρασμα από το inputR στο outputR και με pan ίσης ισχύος στο inputL όπως σε μονοφωνική
περίπτωση.
    outputL = inputL * gainL;
    outputR = inputR + inputL * gainR;
}
```

### HRTF panning (μόνο στερεοφωνικά)

Για HRTF panning απαιτείται μια σειρά από HRTF (Head-related Transfer Function) καταγραφές παλμικών αποκρίσεων σε μια πληθώρα αζιμουθίων και ανυψώσεων. Η υλοποίηση απαιτεί μια ειδικά βελτιστοποιημένη συνάρτηση συνέλιξης. Είναι λίγο πιο ακριβή από την "equalpower", αλλά παρέχει χωρικό ήχο που είναι καλύτερα αντιληπτός.



Εικόνα 5- Διαδικασία του panning μιας πηγής χρησιμοποιώντας HRTF

### Εφέ αποστάσεων

Οι ήχοι που βρίσκονται πιο κοντά ακούγονται και πιο δυνατά, όσο ο ήχος απομακρύνεται τόσο και πιο σιγά ακούγεται. Το πώς ακριβώς η ένταση του ήχου αλλάζει σύμφωνα με την απόσταση από τον ακροατή εξαρτάται από το χαρακτηριστικό του *distanceModel*.

Κατά τη διάρκεια της απόδοσης του ήχου, μια τιμή της απόστασης θα υπολογιστεί με βάση τις θέσεις του panner και του ακροατή σύμφωνα με τα εξής:

```

function dotProduct(v1, v2) {
  var d = 0;
  for (var i = 0; i < Math.min(v1.length, v2.length); i++)
    d += v1[i] * v2[i];
  return d;
}
var v = panner.position - listener.position;
var distance = Math.sqrt(dotProduct(v, v));
  
```

Ύστερα το *distance* θα χρησιμοποιηθεί για τον υπολογισμό του *distanceGain* που εξαρτάται από το χαρακτηριστικό του *distanceModel*. Βλ. την ενότητα **Διεπαφή PannerNode** για περισσότερες πληροφορίες στο πώς υπολογίζεται για κάθε *distanceModel*. Οι τιμές υπολογισμού των εξισώσεων του *distanceModel* περιορίζονται στο εύρος [0, 1].

Σαν μέρος της επεξεργασίας, το PannerNode κλιμακώνει/ πολλαπλασιάζει το ηχητικό σήμα εισόδου με το *distanceGain* για να κάνει τους απόμακρους ήχους πιο ήσυχους και τους πιο κοντινούς δυνατώτερους.

### Ηχητικοί Κώνοι (Sound Cones)

Ο ακροατής και κάθε πηγή ήχου έχουν ένα διάνυσμα προσανατολισμού που περιγράφει με ποιο τρόπο βλέπει ο ένας τον άλλο. Τα χαρακτηριστικά της ηχητικής προβολής κάθε πηγής ήχου περιγράφονται από ένα εσωτερικό και εξωτερικό "κώνο" που περιγράφει την ένταση του ήχου ως συνάρτηση της γωνίας πηγής/ ακροατή από το διάνυσμα προσανατολισμού της πηγής. Έτσι, μία ηχητική πηγή που δείχνει απευθείας στον ακροατή θα είναι πιο δυνατή από ό, τι αν ήταν στραμμένη εκτός άξονα. Οι ηχητικές πηγές μπορούν επίσης να είναι πολυκατευθυντικές (omni-directional).

Ο παρακάτω αλγόριθμος πρέπει να χρησιμοποιηθεί για τον υπολογισμό της συμβολής του gain λόγω της επίδρασης κώνου, με δεδομένη την πηγή (το PannerNode) και τον ακροατή:

```
function dotProduct(v1, v2) {
  var d = 0;
  for (var i = 0; i < Math.min(v1.length, v2.length); i++)
    d += v1[i] * v2[i];
  return d;
}
```

```
function diff(v1, v2) {
  var v = [];
  for (var i = 0; i < Math.min(v1.length, v2.length); i++)
    v[i] = v1[i] - v2[i];
  return v;
}
```

```
function coneGain() {
  if (dotProduct(source.orientation, source.orientation) == 0 || ((source.coneInnerAngle ==
    360) && (source.coneOuterAngle == 360)))
    return 1; // δεν προσδιορίστηκε κώνος - unity gain
```

```
// Ομαλοποίηση διανύσματος πηγής- ακροατή
var sourceToListener = diff(listener.position, source.position);
sourceToListener.normalize();
```

```
var normalizedSourceOrientation = source.orientation;
normalizedSourceOrientation.normalize();

// Γωνία μεταξύ διανύσματος προσανατολισμού της πηγής και του διανύσματος πηγής-ακροατή
var dotProduct = dotProduct(sourceToListener, normalizedSourceOrientation);
var angle = 180 * Math.acos(dotProduct) / Math.PI;
var absAngle = Math.abs(angle);

// Διαίρεση με το 2 μιας και το API είναι με ολόκληρη γωνία (όχι μισή γωνία)
var absInnerAngle = Math.abs(source.coneInnerAngle) / 2;
var absOuterAngle = Math.abs(source.coneOuterAngle) / 2;
var gain = 1;

if (absAngle <= absInnerAngle) {
  // Χωρίς εξασθένηση
  gain = 1;
} else if (absAngle >= absOuterAngle) {
  // Μέγιστη εξασθένηση
  gain = source.coneOuterGain;
} else {
  // Ανάμεσα σε εσωτερικούς και εξωτερικούς κώνους
  // εσωτερικός -> εξωτερικός, το x κυμαίνεται από 0 -> 1
  var x = (absAngle - absInnerAngle) / (absOuterAngle - absInnerAngle);
  gain = (1 - x) + source.coneOuterGain * x;
}
return gain;
}
```

### 3.1.2.2 Διεπαφή *PannerNode*

Αυτή η διεπαφή αναπαριστά έναν κόμβο (node) επεξεργασίας ο οποίος τοποθετεί/χωροθετεί μια εισερχόμενη ροή ήχου στον τρισδιάστατο χώρο. Η χωρικοποίηση είναι σχετική με το AudioListener του AudioContext (listener attribute).

numberOfInputs : 1

numberOfOutputs : 1

channelCount = 2;

channelCountMode = "clamped-max";

channelInterpretation = "speakers";

Η είσοδος αυτού του κόμβου node είναι ή μονοφωνική (1 κανάλι) ή στερεοφωνική (2 κανάλια) και δεν μπορεί να αυξηθεί. Οι συνδέσεις από κόμβους με λιγότερα η περισσότερα κανάλια θα αναμειχθούν ή συνενωθούν καταλλήλως, αλλά ένα `NotSupportedException` πρέπει να εμφανιστεί αν γίνει προσπάθεια να οριστεί αριθμός καναλιών σε τιμή μεγαλύτερη από τα 2 ή αν το `channelCountMode` έχει οριστεί σε "max".

Η έξοδος αυτού του κόμβου είναι αυστηρά κωδικοποιημένη στο στερεοφωνικό (2 κανάλια) και δεν μπορεί να ρυθμιστεί.

Η απαρίθμηση του `PanningModelType` καθορίζει ποιος αλγόριθμος χωροθέτησης θα πρέπει να χρησιμοποιηθεί για την τοποθέτηση του ήχου σε 3D χώρο. Η προεπιλογή είναι το "equalpower".

Αυτός ο κόμβος μπορεί να έχει μια αναφορά σε tail-time. Εάν το `panningModel` έχει οριστεί σε "HRTF", ο κόμβος θα παράγει μη σιωπηλή έξοδο για αθόρυβη είσοδο, λόγω της εγγενούς επεξεργασίας για τις ανταποκρίσεις της κεφαλής.

```
enum PanningModelType {  
    "equalpower",  
    "HRTF"  
};
```

Η απαρίθμηση του `DistanceModelType` καθορίζει ποιος αλγόριθμος θα πρέπει να χρησιμοποιηθεί για την μείωση της έντασης μιας ηχητικής πηγής καθώς απομακρύνεται από τον ακροατή. Η προεπιλογή είναι το "inverse".

Στην περιγραφή κάθε μοντέλου απόστασης παρακάτω, σαν  $d$  θεωρείται η απόσταση ανάμεσα στον ακροατή και τον panner,  $d_{ref}$  είναι η τιμή του `refDistance`,  $d_{max}$  είναι η τιμή του `maxDistance`, και  $f$  είναι η τιμή του `rolloffFactor`.

```
enum DistanceModelType {  
    "linear",  
    "inverse",  
    "exponential"  
};
```

A linear distance model which calculates *distanceGain* according to:

linear

$$1 - f \frac{\max(\min(d, d_{max}), d_{ref}) - d_{ref}}{d_{max} - d_{ref}}$$

That is, *d* is clamped to the interval  $[d_{ref}, d_{max}]$ .

An inverse distance model which calculates *distanceGain* according to:

inverse

$$\frac{d_{ref}}{d_{ref} + f(\max(d, d_{ref}) - d_{ref})}$$

That is, *d* is clamped to the interval  $[d_{ref}, \infty)$ .

An exponential distance model which calculates *distanceGain* according to:

exponential

$$\left( \frac{\max(d, d_{ref})}{d_{ref}} \right)^{-f}$$

That is, *d* is clamped to the interval  $[d_{ref}, \infty)$ .

Εικόνα 6- Αλγόριθμοι του DistanceModelType

### Χαρακτηριστικά/ Ιδιότητες (Attributes)

**coneInnerAngle** τύπου double

Μία παράμετρος για τις κατευθυντήριες πηγές ήχου, είναι μια γωνία, σε μοίρες, στο εσωτερικό της οποίας δεν θα υπάρχει μείωση της έντασης. Η προκαθορισμένη τιμή είναι 360. Η συμπεριφορά είναι απροσδιόριστη εάν η γωνία είναι εκτός του διαστήματος  $[0, 360]$ .

**coneOuterAngle** τύπου double

Μία παράμετρος για τις κατευθυντήριες πηγές ήχου, είναι μια γωνία, σε μοίρες, στο εξωτερικό της οποίας η ένταση θα μειώνεται σε μια σταθερή τιμή του `coneOuterGain`. Η προκαθορισμένη τιμή είναι 360. Η συμπεριφορά είναι απροσδιόριστη εάν η γωνία είναι εκτός του διαστήματος [0, 360].

**coneOuterGain** τύπου `double`

Μία παράμετρος για τις κατευθυντήριες πηγές ήχου, αυτό είναι το `gain` στο εξωτερικό του `coneOuterAngle`. Η προκαθορισμένη τιμή είναι 0. Είναι μια γραμμική τιμή (όχι dB) μέσα στο εύρος [0, 1]. Ένα `InvalidStateError` πρέπει να εμφανιστεί αν η παράμετρος είναι εκτός αυτού του εύρους.

**distanceModel** τύπου `DistanceModelType`

Καθορίζει το μοντέλο απόστασης που χρησιμοποιείται από τον `PannerNode`. Η προκαθορισμένη επιλογή είναι "inverse".

**maxDistance** τύπου `double`

Η μέγιστη απόσταση μεταξύ της πηγής και του ακροατή, μετά την οποία η ένταση δεν θα μειωθεί περαιτέρω. Η προκαθορισμένη τιμή είναι 10000.

**orientationX** τύπου `AudioParam`, μόνο για ανάγνωση

Περιγράφει την X συνιστώσα του διανύσματος της κατεύθυνσης, η πηγή ήχου δείχνει σε 3D καρτεσιανό σύστημα συντεταγμένων. Ανάλογα με το πόσο κατευθυντήριος είναι ο ήχος (ελέγχεται από τα χαρακτηριστικά του κώνου), ένας ήχος που δείχνει μακριά από τον ακροατή μπορεί να είναι πολύ ήσυχος ή εντελώς αθόρυβος. Η προκαθορισμένη τιμή είναι 1. Αυτή η παράμετρος είναι *a-rate* όταν το `panningModel` είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**orientationY** τύπου `AudioParam`, μόνο για ανάγνωση

Περιγράφει την Y συνιστώσα του διανύσματος της κατεύθυνσης, η πηγή ήχου δείχνει σε 3D καρτεσιανό σύστημα συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το `panningModel` είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**orientationZ** τύπου `AudioParam`, μόνο για ανάγνωση

Περιγράφει την Z συνιστώσα του διανύσματος της κατεύθυνσης, η πηγή ήχου δείχνει σε 3D καρτεσιανό σύστημα συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το `panningModel` είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**panningModel** τύπου `PanningModelType`

Καθορίζει το μοντέλο `panning` που χρησιμοποιείται από τον `PannerNode`. Η προκαθορισμένη επιλογή είναι "equalpower".

**positionX** τύπου `AudioParam`, μόνο για ανάγνωση

Ρυθμίζει την X συνιστώσα της θέσης συντεταγμένων της πηγής ήχου σε ένα 3D καρτεσιανό σύστημα. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**positionY** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει την Y συνιστώσα της θέσης συντεταγμένων της πηγής ήχου σε ένα 3D καρτεσιανό σύστημα. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**positionZ** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει την Z συνιστώσα της θέσης συντεταγμένων της πηγής ήχου σε ένα 3D καρτεσιανό σύστημα. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**refDistance** τύπου *double*

Μια αναφορά της απόστασης για τη μείωση της έντασης καθώς η πηγή κινείται πιο μακριά από τον ακροατή. Η προκαθορισμένη τιμή είναι 1.

**rolloffFactor** τύπου *double*

Περιγράφει πόσο γρήγορα μειώνεται η ένταση καθώς η πηγή απομακρύνεται από τον ακροατή. Η προκαθορισμένη τιμή είναι 1.

## Μέθοδοι (Methods)

### setOrientation

Αυτή η μέθοδος έχει καταργηθεί. Είναι ισοδύναμη με τη ρύθμιση του *orientationX*, *orientationY*, και *orientationZ* του *AudioParam* απευθείας.

Περιγράφει σε ποια κατεύθυνση η πηγή ήχου είναι στραμμένη στο 3D καρτεσιανό χώρο συντεταγμένων. Ανάλογα με το πόσο κατευθυντήριος είναι ο ήχος (ελέγχεται από τα χαρακτηριστικά του κώνου), ένας ήχος που δείχνει μακριά από τον ακροατή μπορεί να είναι πολύ ήσυχος ή εντελώς αθόρυβος.

Οι *x*, *y*, *z* παράμετροι αντιπροσωπεύουν ένα διάνυσμα κατεύθυνσης στο 3D χώρο.

Η προκαθορισμένη τιμή είναι (1,0,0)

Parameter	Type	Nullable	Optional
<i>x</i>	<i>float</i>	X	X
<i>y</i>	<i>float</i>	X	X
<i>z</i>	<i>float</i>	X	X

Εικόνα 7- Παράμετροι του *setOrientation* της Διεπαφής *PannerNode*



## setPosition

Αυτή η μέθοδος έχει καταργηθεί. Είναι ισοδύναμη με τη ρύθμιση του positionX, positionY, και positionZ του AudioParam απευθείας.

Ρυθμίζει τη θέση της πηγής ήχου σε σχέση με το χαρακτηριστικό του ακροατή. Χρησιμοποιείται ένα 3D καρτεσιανό σύστημα συντεταγμένων.

Οι x, y, z παράμετροι αντιπροσωπεύουν ένα διάνυσμα κατεύθυνσης στο 3D χώρο.

Η προκαθορισμένη τιμή είναι (0,0,0)

Parameter	Type	Nullable	Optional
x	float	X	X
y	float	X	X
z	float	X	X

Εικόνα 8- Παράμετροι του setPosition της Διεπαφής PannerNode

## PannerOptions

Αυτή η ενότητα καθορίζει τις επιλογές για την κατασκευή του PannerNode. Όλα τα μέλη είναι προαιρετικά· αν δεν έχουν οριστεί, η κανονική προεπιλογή χρησιμοποιείται στην κατασκευή του κόμβου.

```
dictionary PannerOptions : AudioNodeOptions {
    PanningModelType panningModel;
    DistanceModelType distanceModel;
    float positionX;
    float positionY;
    float positionZ;
    float orientationX;
    float orientationY;
    float orientationZ;
    double refDistance;
    double maxDistance;
    double rolloffFactor;
    double coneInnerAngle;
    double coneOuterAngle;
    double coneOuterGain;
};
```

Εικόνα 9- Μέλη του PannerOptions

## Λεξικό μελών του PannerOptions

**coneInnerAngle** τύπου double

Η αρχική τιμή για το χαρακτηριστικό του `coneInnerAngle` του κόμβου.

**coneOuterAngle** τύπου `double`

Η αρχική τιμή για το χαρακτηριστικό του `coneOuterAngle` του κόμβου.

**coneOuterGain** τύπου `double`

Η αρχική τιμή για το χαρακτηριστικό του `coneOuterGain` του κόμβου.

**distanceModel** τύπου *DistanceModelType*

Το μοντέλο απόστασης που θα χρησιμοποιηθεί για τον κόμβο.

**maxDistance** τύπου `double`

Η αρχική τιμή για το χαρακτηριστικό του `maxDistance` του κόμβου.

**orientationX** τύπου `float`

Η αρχική τιμή του X για το `orientationX` του `AudioParam`.

**orientationY** τύπου `float`

Η αρχική τιμή του The initial Y για το `orientationY` του `AudioParam`.

**orientationZ** τύπου `float`

Η αρχική τιμή του Z για το `orientationZ` του `AudioParam`.

**panningModel** τύπου *PanningModelType*

Το μοντέλο panning που θα χρησιμοποιηθεί για τον κόμβο.

**positionX** τύπου `float`

Η αρχική τιμή του X για το `positionX` του `AudioParam`.

**positionY** τύπου `float`

Η αρχική τιμή του Y για το `positionY` του `AudioParam`.

**positionZ** τύπου `float`

Η αρχική τιμή του Z για το `positionZ` του `AudioParam`.

**refDistance** τύπου `double`

Η αρχική τιμή για το χαρακτηριστικό του `refDistance` για τον κόμβο.

**rolloffFactor** τύπου `double`

Η αρχική τιμή για το χαρακτηριστικό του `rolloffFactor` για τον κόμβο.

### 3.1.2.3 Διεπαφή *AudioListener*

Αυτή η διεπαφή αναπαριστά την θέση και τον προσανατολισμό του ατόμου που ακούει στη σκηνή. Όλα τα αντικείμενα του *PannerNode* χωροθετούνται σε σχέση με τον ακροατή του *BaseAudioContext*. Βλ. την ενότητα Χωρικοποίηση (Spatialization/Panning) για περισσότερες λεπτομέρειες σχετικά με την χωρικοποίηση.

Οι παράμετροι *positionX*, *positionY*, *positionZ* αναπαριστούν την θέση του ακροατή σε 3D καρτεσιανό σύστημα συντεταγμένων. Τα αντικείμενα του *PannerNode* χρησιμοποιούν αυτή τη θέση σε σχέση με την κάθε πηγή ήχου ξεχωριστά για την χωροθέτηση.

Οι παράμετροι *forwardX*, *forwardY*, *forwardZ* αναπαριστούν ένα διάνυσμα κατεύθυνσης στο 3D χώρο. Ένα *forward* διάνυσμα και ένα *up* διάνυσμα χρησιμοποιούνται για τον προσδιορισμό του προσανατολισμού του ακροατή. Με απλά λόγια, το διάνυσμα *forward* αντιπροσωπεύει την κατεύθυνση που δείχνει η μύτη του ατόμου. Το διάνυσμα *up* αντιπροσωπεύει την κατεύθυνση που δείχνει η κορυφή του κεφαλιού του ατόμου. Αυτές οι τιμές αναμένεται να είναι γραμμικά ανεξάρτητες (σε ορθή γωνία μεταξύ τους). Για τις κανονιστικές προδιαγραφές στο πώς πρέπει να ερμηνεύονται αυτές οι τιμές, βλ. την ενότητα Χωρικοποίηση (Spatialization/Panning).

```
interface AudioListener {
  readonly attribute AudioParam positionX;
  readonly attribute AudioParam positionY;
  readonly attribute AudioParam positionZ;
  readonly attribute AudioParam forwardX;
  readonly attribute AudioParam forwardY;
  readonly attribute AudioParam forwardZ;
  readonly attribute AudioParam upX;
  readonly attribute AudioParam upY;
  readonly attribute AudioParam upZ;
  void setPosition (float x, float y, float z);
  void setOrientation (float x, float y, float z, float xUp, float yUp, float zUp);
};
```

Εικόνα 10- Χαρακτηριστικά/ Ιδιότητες της Διεπαφής *AudioListener*

#### Χαρακτηριστικά/ Ιδιότητες (Attributes)

**forwardX** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα X της εμπρόσθιας κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**forwardY** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα Y της εμπρόσθιας κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**forwardZ** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα Z της εμπρόσθιας κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**positionX** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη X της θέσης του ακροατή σε 3D καρτεσιανό επίπεδο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**positionY** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη Y της θέσης του ακροατή σε 3D καρτεσιανό επίπεδο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**positionZ** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη Z της θέσης του ακροατή σε 3D καρτεσιανό επίπεδο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate* όταν το *panningModel* του *PannerNode* είναι "equalpower" και *k-rate* σε άλλη περίπτωση.

**upX** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα X της άνω κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate*.

**upY** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα Y της άνω κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 1. Αυτή η παράμετρος είναι *a-rate*.

**upZ** τύπου *AudioParam*, μόνο για ανάγνωση

Ρυθμίζει τη συντεταγμένη συνιστώσα Z της άνω κατεύθυνσης, ο ακροατής δείχνει σε 3D καρτεσιανό χώρο συντεταγμένων. Η προκαθορισμένη τιμή είναι 0. Αυτή η παράμετρος είναι *a-rate*.

## Μέθοδοι (Methods)

### setOrientation

Αυτή η μέθοδος έχει καταργηθεί. Είναι ισοδύναμη με τη ρύθμιση του *orientationX.value*, *orientationY.value*, *orientationZ.value*, *upX.value*, *upY.value*, και *upZ.value* απευθείας με δεδομένες τις τιμές *x*, *y*, *z*, *xUp*, *yUp*, και *zUp* αντίστοιχα.

Περιγράφει σε ποια κατεύθυνση ο ακροατής είναι στραμμένος στο 3D καρτεσιανό επίπεδο συντεταγμένων. Παρέχεται το διάνυσμα *front* αλλά και το διάνυσμα *up*. Με απλά λόγια, το

διάνυσμα *front* αντιπροσωπεύει την κατεύθυνση που δείχνει η μύτη του ατόμου. Το διάνυσμα *up* αντιπροσωπεύει την κατεύθυνση που δείχνει η κορυφή του κεφαλιού του ατόμου. Αυτές οι τιμές αναμένεται να είναι γραμμικά ανεξάρτητες (σε ορθή γωνία μεταξύ τους). Για τις κανονιστικές προδιαγραφές στο πώς πρέπει να ερμηνεύονται αυτές οι τιμές, Βλ. την ενότητα Χωρικοποίηση (Spatialization/Panning).

Οι παράμετροι *x*, *y*, *z* αντιπροσωπεύουν ένα *front* διάνυσμα κατεύθυνσης σε 3D χώρο, με τις προκαθορισμένες τιμές να είναι (0,0,-1).

Οι παράμετροι *xUp*, *yUp*, *zUp* αντιπροσωπεύουν ένα *up* διάνυσμα κατεύθυνσης σε 3D χώρο, με τις προκαθορισμένες τιμές να είναι (0,1,0).

Parameter	Type	Nullable	Optional
<i>x</i>	float	X	X
<i>y</i>	float	X	X
<i>z</i>	float	X	X
<i>xUp</i>	float	X	X
<i>yUp</i>	float	X	X
<i>zUp</i>	float	X	X

Εικόνα 11- Παράμετροι του `setOrientation` της Διεπαφής `AudioListener`

### **setPosition**

Αυτή η μέθοδος έχει καταργηθεί. Είναι ισοδύναμη με τη ρύθμιση του `positionX.value`, `positionY.value`, και `positionZ.value` απευθείας με δεδομένες τις τιμές *x*, *y*, και *z values*, αντίστοιχα.

Ρυθμίζει τη θέση του ακροατή σε 3D καρτεσιανό επίπεδο συντεταγμένων. Τα αντικείμενα του `PannerNode` χρησιμοποιούν αυτή τη θέση σχετικά με την κάθε πηγή ήχου ξεχωριστά για χωροθέτηση.

Οι παράμετροι *x*, *y*, *z* αντιπροσωπεύουν τις συντεταγμένες σε 3D χώρο.

Οι προκαθορισμένες τιμές είναι (0,0,0)

Parameter	Type	Nullable	Optional
<i>x</i>	float	X	X
<i>y</i>	float	X	X
<i>z</i>	float	X	X

Εικόνα 12- Παράμετροι του `setPosition` της Διεπαφής `AudioListener`

### 3.1.2.4 Διεπαφή *StereoPannerNode*

Αυτή η διεπαφή αναπαριστά έναν κόμβο (node) επεξεργασίας ο οποίος τοποθετεί μια εισερχόμενη ηχητική ροή σε στερεοφωνική εικόνα χρησιμοποιώντας χαμηλού κόστους αλγόριθμο panning ίσης ισχύος (Equal-power panning). Αυτό το εφέ panning είναι κοινό στην τοποθέτηση στοιχείων ήχου σε μια στερεοφωνική ροή.

numberOfInputs : 1

numberOfOutputs : 1

channelCount = 2;

channelCountMode = "clamped-max";

channelInterpretation = "speakers";

Η είσοδος αυτού του κόμβου είναι στερεοφωνική (2 κανάλια) και δεν μπορεί να αυξηθεί. Συνδέσεις από κόμβους με λιγότερα ή περισσότερα κανάλια θα αναμειχθούν ή συνενωθούν καταλλήλως, αλλά ένα `NotSupportedError` πρέπει να εμφανιστεί αν γίνει προσπάθεια να οριστεί αριθμός καναλιών σε τιμή μεγαλύτερη από τα 2 ή αν το `channelCountMode` έχει οριστεί σε "max".

Η έξοδος αυτού του κόμβου είναι αυστηρά κωδικοποιημένη στο στερεοφωνικό (2 κανάλια) και δεν μπορεί να ρυθμιστεί.

Αυτός ο κόμβος μπορεί να έχει μια αναφορά σε `tail-time`.

[Constructor(BaseAudioContext context, optional StereoPannerOptions options)]

```
interface StereoPannerNode : AudioNode {
  readonly attribute AudioParam pan;
};
```

#### Χαρακτηριστικά/ Ιδιότητες (Attributes)

**pan** τύπου *AudioParam*, μόνο για ανάγνωση

Η θέση της εισόδου στην έξοδο της στερεοφωνικής εικόνας. Το -1 αντιπροσωπεύει το αριστερό κανάλι, το +1 αντιπροσωπεύει το δεξί κανάλι. Η προκαθορισμένη τιμή είναι 0, και το σταθερό εύρος (nominal range) είναι [-1, 1]. Αυτή η παράμετρος είναι *a-rate*.

#### StereoPannerOptions

Αυτή η ενότητα καθορίζει τις επιλογές για την κατασκευή του `StereoPannerNode`. Όλα τα μέλη είναι προαιρετικά: αν δεν έχουν οριστεί, η κανονική προεπιλογή χρησιμοποιείται στην κατασκευή του κόμβου.

```
dictionary StereoPannerOptions : AudioNodeOptions {
```

```
float pan;  
};
```

### Λεξικό μελών του StereoPannerOptions

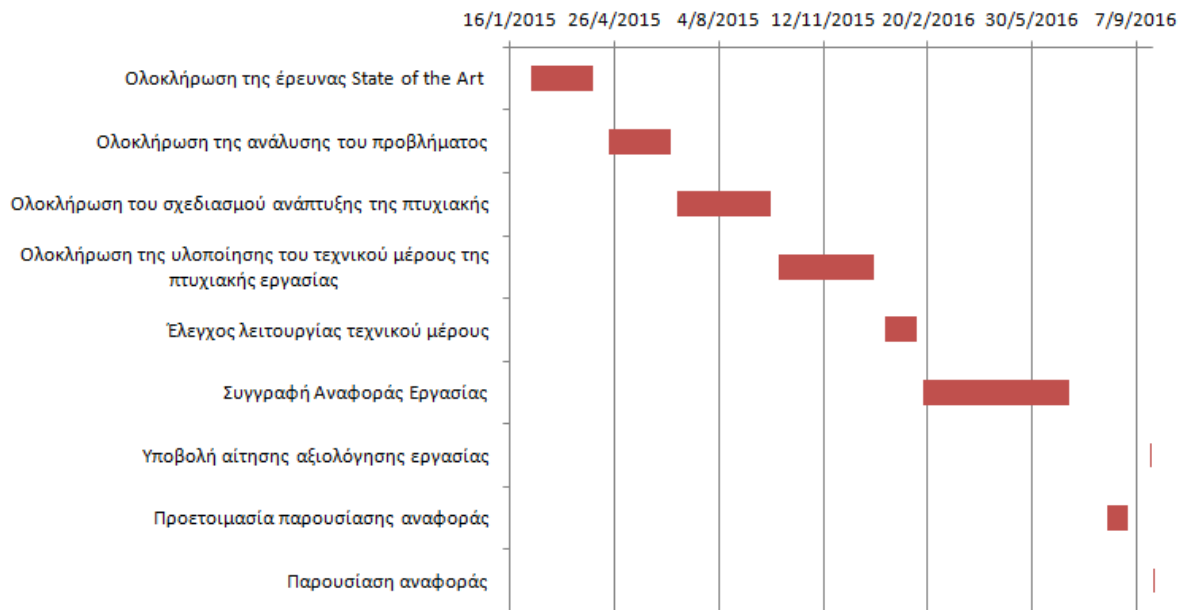
**pan** τύπου float

Η αρχική τιμή του *pan* του AudioParam.

## 3.2 Σημαντικοί στόχοι για την ολοκλήρωση της πτυχιακής

Ολοκλήρωση της έρευνας State of the Art	60
Ολοκλήρωση της ανάλυσης του προβλήματος	60
Ολοκλήρωση του σχεδιασμού ανάπτυξης της πτυχιακής	90
Ολοκλήρωση της υλοποίησης του τεχνικού μέρους της πτυχιακής εργασίας	90
Έλεγχος λειτουργίας τεχνικού μέρους	30
Συγγραφή Αναφοράς Εργασίας	140
Υποβολή αίτησης αξιολόγησης εργασίας	1
Προετοιμασία παρουσίασης αναφοράς	20
Παρουσίαση αναφοράς	1

### 3.2.1 Χρονοδιάγραμμα Εργασίας (Gantt chart)



Εικόνα 13- Χρονοδιάγραμμα Εργασίας (Gantt chart)



## 4 Σύστημα εικονικής αναπαράστασης του ήχου σε 3D χώρο

### 4.1 Σχεδιασμός υλοποίησης

Αυτή η ενότητα είναι αφιερωμένη στην εφαρμογή του προτεινόμενου σχεδίου. Στην πρώτη θέση είναι η εγγραφή του Web Audio API και των προσαρμοσμένων στοιχείων στο πλαίσιο του X3DOM. Όλοι αυτοί οι κόμβοι χρησιμοποιούνται από το HTML DOM, το οποίο είναι ουσιαστικά μια σκηνή του X3D, απευθείας από το X3DOM και όχι μέσα από τη δομή JavaScript. Από την άλλη πλευρά, ο κώδικας JavaScript, ο οποίος υποδεικνύεται ως JavaScript Controller, παραχωρεί όλη την αναγκαία οδηγία του σχεδιασμού και του ρόλου των κόμβων. Αλληλεπιδρά με το HTML αρχείο, με σκοπό την ανάλυση της 3D σκηνής και ανανεώνεται σε κάθε πιθανή αλλαγή στη σκηνή. Στις επόμενες ενότητες, περιγράφεται μια πιο λεπτομερής περιγραφή στην οποία οι κόμβοι εισέρχονται στον πυρήνα του X3DOM προκειμένου να αναγνωριστούν και να χρησιμοποιηθούν ως στοιχεία του X3DOM, καθώς και η διαδικασία του JavaScript Controller.

#### 4.1.1 Web Audio API/ Εγγραφή ενισχυμένων X3D κόμβων στο X3DOM

Τα χαρακτηριστικά του AudioClip (X3D κόμβος) και Sound (X3D κόμβος) έχουν επανεισαχθεί στο X3DOM με τη χρήση:

- Δύο ενισχυμένων κόμβων X3D
  1. AudioSound
  2. AudioSource
- Μια προστιθέμενη καθορισμένη σειρά κόμβων του Web Audio API.

Το πρώτο στάδιο της εργασίας μας είναι η εγγραφή των νέων στοιχείων στο πυρήνα του X3DOM. Πέραν αυτού ανταλλάσσει αμοιβαία πληροφορίες με το στοιχείο AudioBufferSourceNode του Web Audio API. Για οπτική αναπαράσταση της δομής υλοποίησης, ο αναγνώστης παραπέμπεται στο σχ. 3.

#### 4.1.2 Εγγεγραμμένοι κόμβοι του Web Audio API στο X3DOM

Ένα σύνολο κόμβων του Web Audio API καταχωρήθηκαν στο X3DOM, προκειμένου να εφαρμοστεί το προτεινόμενο έργο. Για την απαρίθμηση, αξιοποιήσαμε το AudioDestinationNode και το AudioBufferSourceNode, που αντιπροσωπεύουν τον τελικό προορισμό του ήχου και την πηγή ήχου που αποτελείται από ηχητικά δεδομένα στη μνήμη.

Ομοίως, το GainNode επηρεάζει την ένταση του ήχου και το BiquadFilterNode υποστηρίζει όλους τους συνηθισμένους τύπους φίλτρου δεύτερης τάξης. Επιπλέον, το DelayNode προκαλεί

καθυστέρηση μεταξύ της άφιξης ενός δεδομένου εισόδου και την αναπαραγωγή του στην έξοδο. Το AudioListener αντιπροσωπεύει τη θέση ενός ατόμου ακούγοντας μια πηγή ήχου σε 3D χώρο και κάθε πηγή μπορεί να περάσει μέσα από ένα PannerNode, το οποίο δίνει την αίσθηση χωρικοποίησης του εισερχόμενου ήχου. Με βάση την σχετική θέση των πηγών και του ακροατή, οι σωστές τροποποιήσεις του Gain μπορούν να υπολογισθούν με αυτή τη μέθοδο.

Επίσης, καταχωρίσαμε το StereoPannerNode ώστε να μετακινήσουμε μια ροή ήχου προς τα αριστερά ή τα δεξιά. Ένας ακόμη εγγεγραμμένος κόμβος είναι ο ConvolverNode ο οποίος είναι ουσιαστικά ένα πολύ σύνθετο φίλτρο (όπως το BiquadFilterNode), αλλά αντίθετα από το να επιλέγει από ένα σύνολο αποτελεσματικών τύπων, μπορεί να ρυθμιστεί με μια αυθαίρετη απόκριση του φίλτρου. Το AnalyserNode αποσκοπεί στην παροχή συχνότητας σε πραγματικό χρόνο και στην ανάλυση πληροφορίας σε πεδίο χρόνου. Το ChannelSplitterNode χρησιμοποιήθηκε για το διαχωρισμό των διαφορετικών καναλιών της πηγής ήχου και χρησιμοποιείται συχνά σε συνδυασμό με αντίθετο του, το ChannelMergerNode.

Τέλος, συνιστάται η εισαγωγή του DynamicsCompressorNode το οποίο χρησιμοποιήθηκε για τη συμπίεση των εφέ, το WaveShaperNode για την αναπαράσταση ενός μη-γραμμικού παραμορφωτή και το OscillatorNode για την παράδοση μιας περιοδικής κυματομορφής.

Αξίζει να σημειωθεί ότι η κόμβοι StereoPannerNode, ConvolverNode, WaveShaperNode και OscillatorNode έχουν καταχωρηθεί ως επιπλέον κόμβοι, προκειμένου να παρέχουν περαιτέρω λειτουργίες ήχου στη web 3D σκηνή και να επιτύχει μια υψηλότερη διάσταση ρεαλισμού στο 3D περιβάλλον.

### 4.1.3 Ενισχυμένοι κόμβοι X3D

Οι ενισχυμένοι κόμβοι X3D, οι οποίοι αναπτύχθηκαν για τις ανάγκες του προτεινόμενου έργου, είναι το AudioSound και το AudioSource. Το πρώτο είναι το «μητρικό» στοιχείο κάθε νέου στοιχείου ενώ το δεύτερο είναι ένας συνδυασμός του αναπτυξιακού κόμβου σε X3DOM, X3DSoundSourceNode και το σύνολο των χαρακτηριστικών του αντίστοιχου κόμβου X3D. Συγκεκριμένα, η καταχώριση του AudioSound έχει αναπτυχθεί με την εξής δομή:

#### Εγγραφή του AudioSound Node στο X3DOM

```
x3dom.registerNodeType("AudioSound"){
  this.addField_SFNode("transform",x3dom.nodeTypes.Transform),
  this.addField_SFNode("source", x3dom.nodeTypes.AudioSource),
  this.addField_SFNode("panner", x3dom.nodeTypes.PannerNode),
  this.addField_SFNode("filter",x3dom.nodeTypes.BiquadFilterNode),
  this.addField_SFNode("delay", x3dom.nodeTypes.DelayNode),
  this.addField_SFFloat("playbackRate", 1),
  this.addField_SFNode("metadata",x3dom.nodeTypes.X3DMetadataObject)}
```

Όπως αναφέρθηκε παραπάνω, το Audio Sound περιλαμβάνει πεδίο transform το οποίο είναι συνδεδεμένο με το Transform τύπου κόμβου, προκειμένου να συνδεθούν χρησιμοποιώντας το χαρακτηριστικό DEF/ USE. Επιπλέον, το χαρακτηριστικό της πηγής είναι να προσδιορίσει την πηγή ήχου, μέσω του ενισχυμένου κόμβου AudioSource του X3D. Επιπλέον, η ιδιότητα του panner συνδέεται με το PannerNode, το οποίο είναι αξιόπιστο για την αναπαράσταση του θέσης και της συμπεριφοράς της πηγής του ήχου. Μετά από αυτό, το φίλτρο προορίζεται για να εισάγει ένα απλό φίλτρο low-order. Τέλος, τα χαρακτηριστικά του delay και του playbackRate χρησιμοποιούνται για την καθυστέρηση και το ρυθμό εισόδου των ηχητικών δεδομένων. Επιπλέον, το AudioSource έχει καταχωρηθεί με την εξής δομή:

#### Εγγραφή του AudioSource Node στο X3DOM

```
x3dom.registerNodeType("AudioSource"){
  this.addField_SFString("description", ""),
  this.addField_SFBool("loop", !1),
  this.addField_SFNode("metadata",
    x3dom.nodeTypes.X3DMetadataObject),
  this.addField_MFString("url", []),
  this.addField_SFTIME("pauseTime", 0),
  this.addField_SFFloat("pitch", 1),
  this.addField_SFTIME("resumeTime", 0),
  this.addField_SFTIME("startTime", 0),
  this.addField_SFTIME("stopTime", 0) }
```

Το πεδίο description καθορίζει μια κειμενική περιγραφή της πηγής ήχου, το χαρακτηριστικό loop φροντίζει για την επανάληψη της πηγής ήχου. Το πεδίο url δείχνει την τοποθεσία της επιλεγμένης ηχητικής πηγής και το pitch είναι ένας παράγοντας πολλαπλασιασμού που εφαρμόζεται στην δειγματοληψία του ήχου και της αναπαραγωγής. Τέλος, στη δομή του X3D (pauseTime, startTime, stopTime) είχαν την προεπιλεγμένη τιμή (0, 0, 0). Στην εφαρμογή μας, οι αντίστοιχες ιδιότητες υλοποιήθηκαν με τις προεπιλεγμένες τιμές (-1, 0, -1), για την περίπτωση που το pauseTime και το stopTime δεν χρειάζεται να χρησιμοποιηθούν.

#### 4.1.4 JavaScript Controller

Ο ρόλος του JavaScript Controller είναι να μμηθεί τον κόμβο ScriptProcessorNode του Web Audio API, ο οποίος παρέχει τη δυνατότητα της σύνθεσης και της διαχείρισης του web ήχου, απευθείας σε JavaScript. Εξίσου σημαντικό είναι το γεγονός ότι ένα ή περισσότερα στοιχεία AudioSource, το οποίο ενσωματώνει τους κόμβους AudioSource, PannerNode, BiquadFilterNode, DelayNode και Transform, μπορούν να συμπεριληφθούν στο HTML.

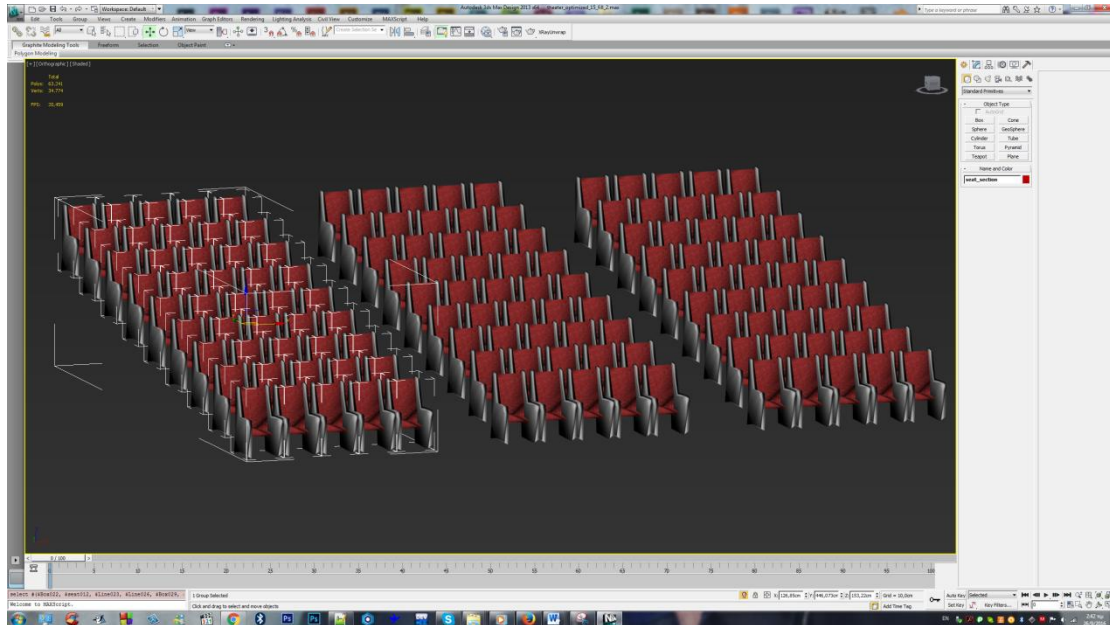
Κατά συνέπεια, η δομή του JavaScript Controller αρχίζει με μια επαναλαμβανόμενη ανίχνευση στο HTML DOM, μέχρι να βρείτε ένα κόμβο AudioSound (ή έναν Inline κόμβο. Στην περίπτωση αυτή, η σκηνή είναι ένα αρχείο X3D και όλοι οι κόμβοι που φορτώνονται από τον κόμβο Inline μέσω του "url" του αρχείου X3D.), ουσιαστικά μέχρι να αναγνωρίσει ένα στοιχείο ήχου του X3DOM. Μόλις αυτό επιτευχθεί, θα δημιουργηθεί ένα νέο αντικείμενο 3D Sound.

Στη συνέχεια, γίνεται επίκληση των καταχωρημένων κόμβων, ακολουθώντας συγκεκριμένη σειρά, AudioSourceNode, AnalyserNode, WaveShaperNode, BiquadFilterNode, ConvolverNode, GainNode, PannerNode και AudioDestinationNode. Οι παραπάνω κόμβοι θα αναγνωριστούν και θα προσαρτηθούν σε μια σειρά, έτσι ώστε να είναι διαθέσιμοι. Μαζί με αυτό, το αντικείμενο 3D Sound αρχικοποιήθηκε και ενημερώθηκε, μέσω της ανταλλαγής δεδομένων με το HTML DOM. Εκτός από το, ArrayBuffer και XMLHttpRequest χρησιμοποιούνται, με σκοπό να φορτώσουν και να παίξουν ήχο. Σε περίπτωση που αλλάξουν οι οπτικές και ακουστικές συνθήκες, το HTML DOM θα ενημερωθεί αντιστοίχως.

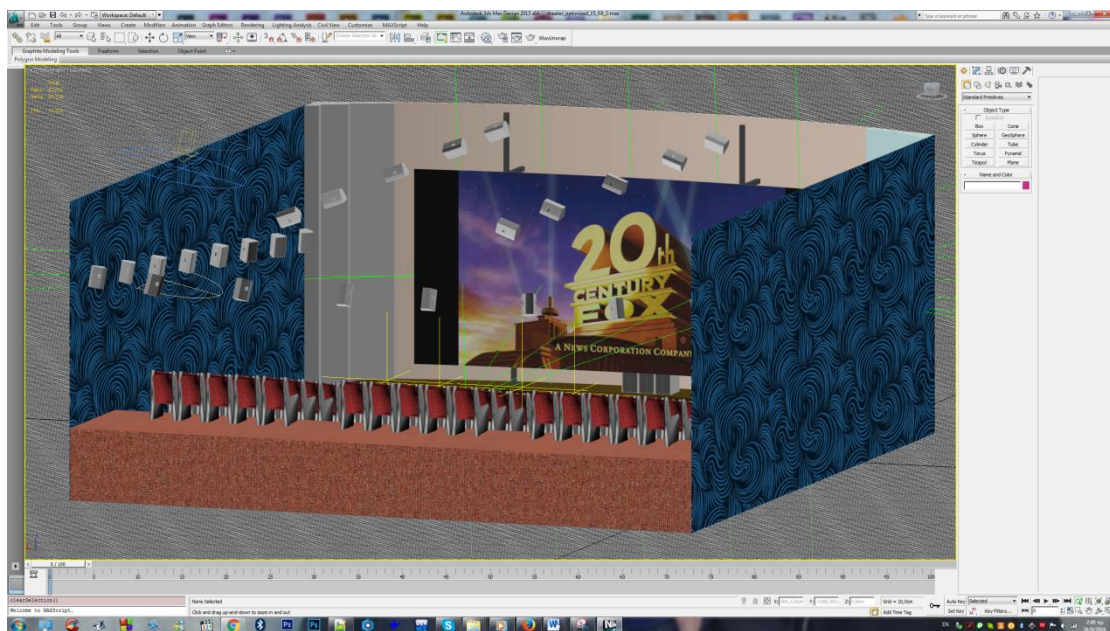
## 4.2 Υλοποίηση

Σε αυτή την ενότητα ακολουθεί η υλοποίηση ενός παραδείγματος στο οποίο χρησιμοποιήθηκαν οι νέοι καταχωρημένοι κόμβοι προκειμένου να επαληθευτεί η εγκυρότητα της μεθόδου μας. Οι δοκιμές έχουν ερευνηθεί με τη χρήση του Google Chrome 41.0, Mozilla Firefox 36.0 και Opera 28.0. Η επιλογή αυτή βασίζεται στο γεγονός ότι αυτά τα προγράμματα περιήγησης υποστηρίζουν το Web Audio API [56]. Στη συνέχεια, ακολουθεί μια περιγραφή η οποία παρουσιάζει το σχεδιασμό αυτού του παραδείγματος λεπτομερώς.

Για τη συγκεκριμένη εργασία δημιουργήσαμε έναν τρισδιάστατο κινηματογράφο με τη χρήση του προγράμματος Autodesk 3DS Max. Ο λόγος που χρησιμοποιήσαμε το συγκεκριμένο πρόγραμμα για την μοντελοποίηση του κινηματογράφου, είναι η δυνατότητα που δίνει το 3DS Max για τρισδιάστατη σχεδίαση σε εύχρηστο περιβάλλον, παρέχοντας ένα πλήθος από επιλογές και εργαλεία που κάνουν την σχεδίαση εύκολη και γρήγορη. Επίσης, το 3DS Max μας δίνει την δυνατότητα εξαγωγής σε X3D και XHTML με την χρήση του InstantExport. Το InstantExport είναι ένας εξαγωγέας InstantReality X3D για το 3DS Max και δεν εξάγει μόνο XML-X3D καθώς και VRML, την κλασική κωδικοποίηση, αλλά μπορεί επίσης να εξάγει απευθείας και HTML/ XHTML αρχεία.

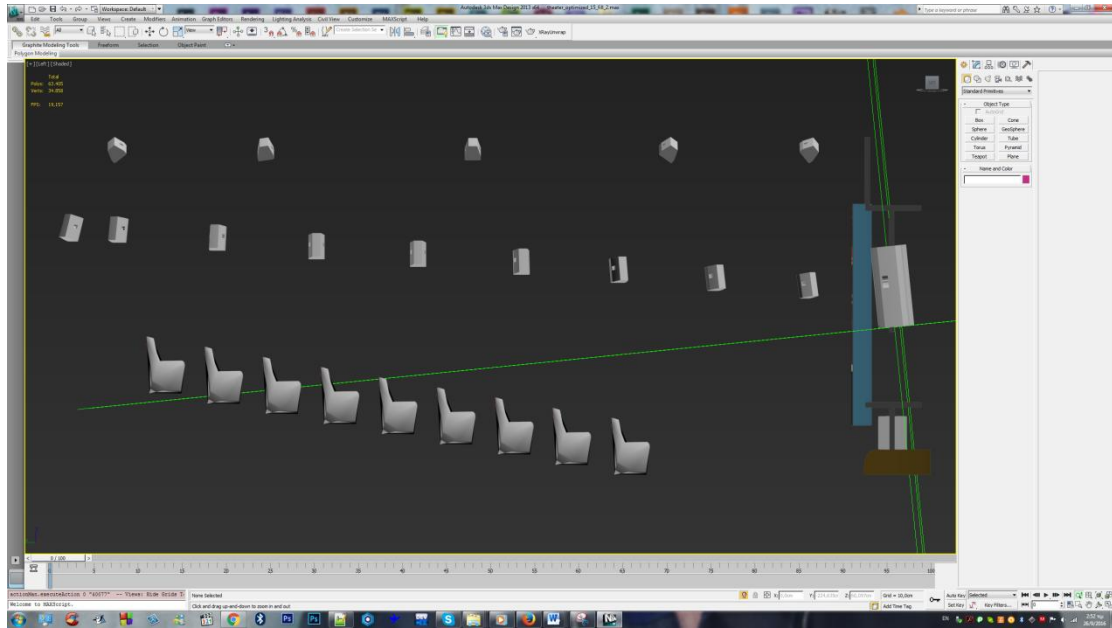


Εικόνα 14- Σχεδιασμός θέσεων του κινηματογράφου στο 3DS Max

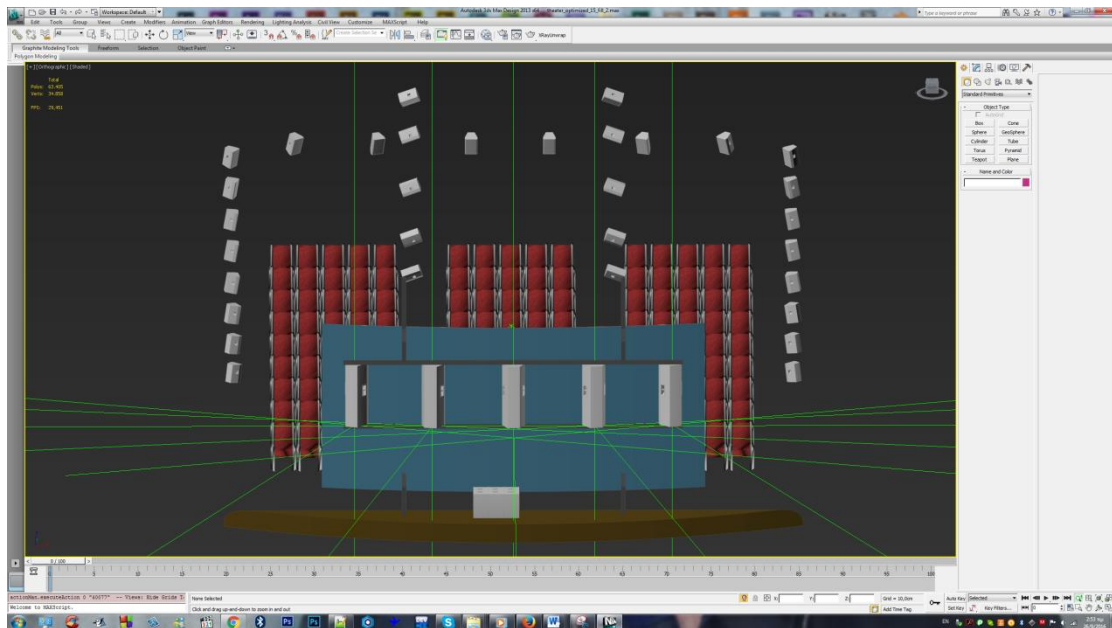


Εικόνα 15- Σχεδιασμός αντικειμένων του κινηματογράφου στο 3DS Max

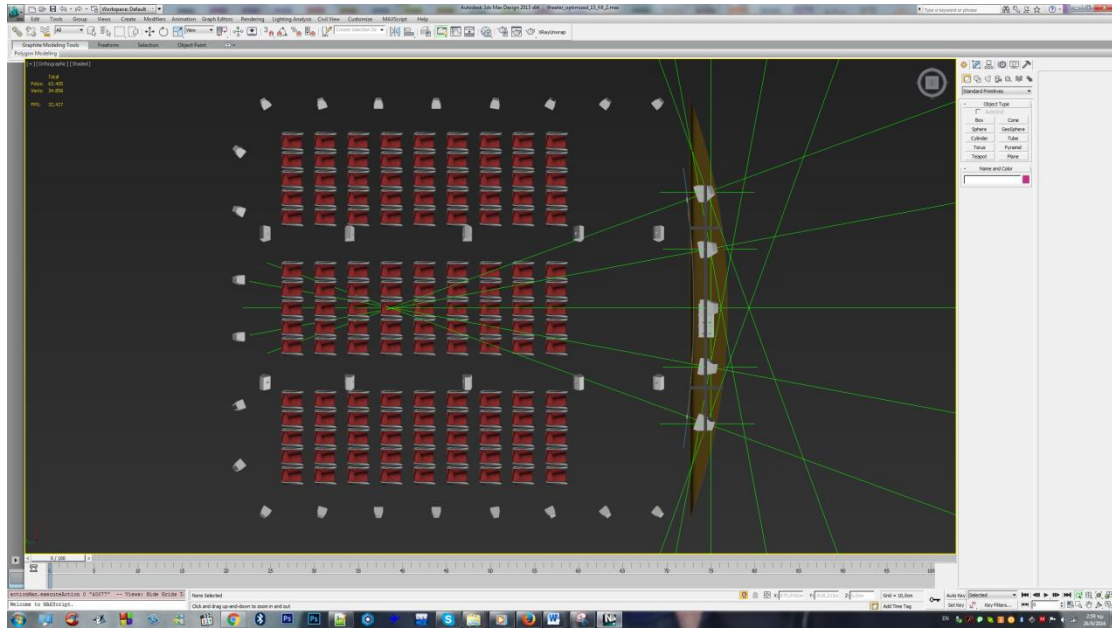
Για την τοποθέτηση των ηχείων στον τρισδιάστατο κινηματογράφο έγινε μελέτη της τεχνολογίας surround ήχου της Dolby Atmos, μόνο για λόγους απεικόνισης. Στις παρακάτω εικόνες παρουσιάζεται η τοποθέτηση των ηχείων από διαφορετικές πλευρές.



Εικόνα 16- Δεξιά όψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max

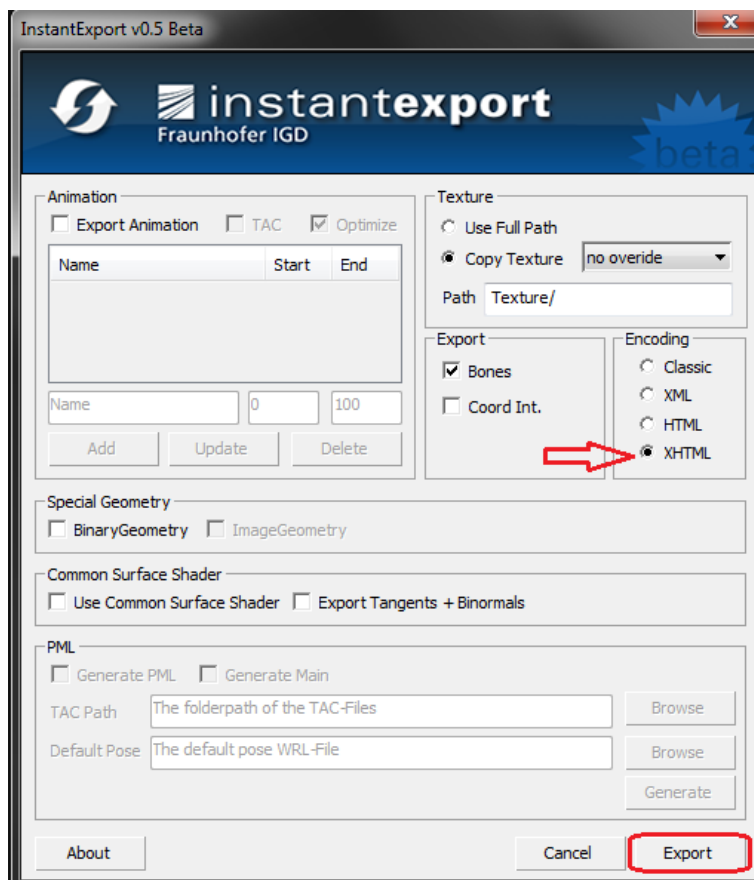
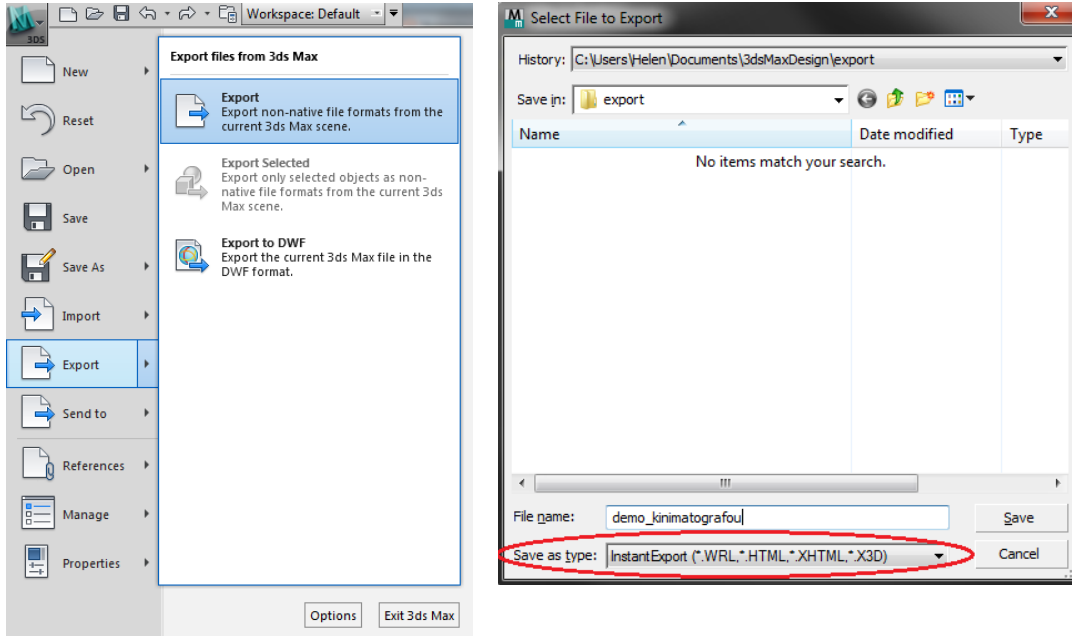


Εικόνα 17- Πάνω-εμπρόσθια όψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max



Εικόνα 18- Κάτοψη της τοποθέτησης ηχείων του κινηματογράφου στο 3DS Max

Μετά την ολοκλήρωση του σχεδιασμού του κινηματογράφου έγινε εξαγωγή σε μορφή ΧΗΤΜL με τη χρήση του InstantExport. Οι παρακάτω εικόνες παρουσιάζουν τα βήματα για την εξαγωγή.



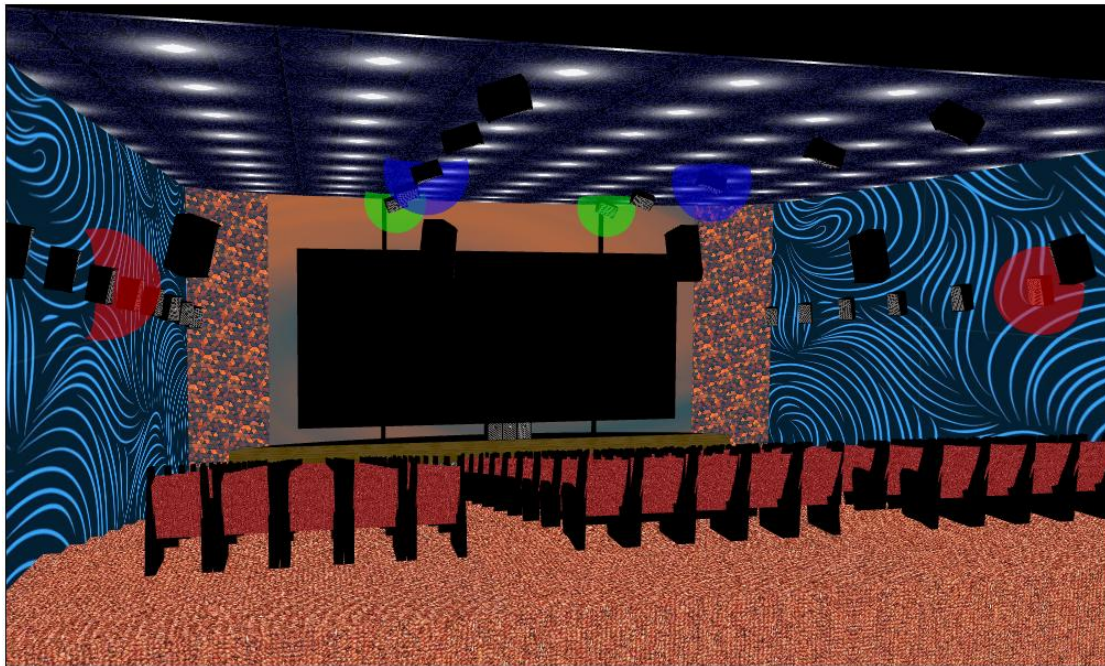
Εικόνα 19- Εξαγωγή σκηνής σε ΧΗΤΜL με τη χρήση του InstantExport



Με την ολοκλήρωση της εξαγωγής έχουμε έτοιμη την X3D σκηνή σε μορφή αρχείου XHTML. Τώρα μπορούμε με τη χρήση των νέων κόμβων που δημιουργήσαμε στο X3DOM (AudioSound και AudioSource) να ενσωματώσουμε στη σκηνή μας ήχο. Για να δημιουργήσουμε το εφέ της χωρικοποίησης θα χρησιμοποιήσουμε και τον κόμβο PannerNode.

Για το συγκεκριμένο παράδειγμα χρησιμοποιήσαμε ένα κομμάτι από μια ταινία μικρού μήκους (<https://goo.gl/xT3HM4>) για να προβάλλεται στην οθόνη του κινηματογράφου. Επιλέξαμε την ταινία αυτή γιατί οι δημιουργοί της παρέχουν ξεχωριστά τα κομμάτια του ήχου, συγκεκριμένα τρία ξεχωριστά κομμάτια ήχου (Διάλογοι, Εφέ και Μουσική) που απαρτίζουν το ηχητικό σύνολο του βίντεο.

Στη συνέχεια χωρίσαμε κάθε κομμάτι σε 2 ξεχωριστά κομμάτια (1 για το δεξί και 1 για το αριστερό κανάλι) με τη χρήση του Adobe Audition και μετά επιλέξαμε σε ποιά ηχεία θα αναπαράγεται το κάθε κομμάτι όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 20- Προσδιορισμός ηχητικών πηγών στη σκηνή

Οι σφαίρες με τα χρώματα δείχνουν ποία ηχεία θα αναπαράγουν τα κομμάτια. Πιο συγκεκριμένα οι 2 πράσινες σφαίρες δείχνουν τα ηχεία που αναπαράγουν το κομμάτι των εφέ της ταινίας, η δεξιά σφαίρα το κομμάτι του δεξιού καναλιού και η αριστερή σφαίρα του αριστερού καναλιού. Οι 2 μπλε σφαίρες δείχνουν τα ηχεία που αναπαράγουν το κομμάτι του διαλόγου της ταινίας, η δεξιά σφαίρα το κομμάτι του δεξιού καναλιού και η αριστερή σφαίρα του αριστερού καναλιού. Τέλος, οι 2 κόκκινες σφαίρες δείχνουν τα ηχεία που αναπαράγουν το κομμάτι της μουσικής της ταινίας, η δεξιά σφαίρα το κομμάτι του δεξιού καναλιού και η αριστερή σφαίρα του αριστερού καναλιού.

Η σύνδεση των ήχων στα ηχεία του κινηματογράφου μέσα στον κώδικα του XHTML αρχείου φαίνεται στην παρακάτω εικόνα.

Βλέπουμε ότι ο κόμβος AudioSound περιλαμβάνει ένα γκρουπ Transform το οποίο περιέχει τους κόμβους PannerNode και AudioSource. Το γκρουπ Transform χρησιμοποιείται για να τοποθετήσει τον ήχο στο ηχείο που βρίσκεται στη σκηνή, με τη χρήση του DEF/ USE. Συνδέοντας έτσι τη θέση της πηγής του ήχου που κάνει USE το id του γκρουπ Transform που περιέχει το ηχείο με το συγκεκριμένο DEF id.

```
<AudioSound playbackRate='1.0'>
<Transform USE='top_left1' />
<PannerNode position='0 0 0' orientation='0 0 1 0' velocity='0 0 0' coneInnerAngle='360'
coneOuterAngle='360' coneOuterGain='0' distanceModel='inverse' maxDistance='10000'
panningModel='HRTF' refDistance='1' rolloffFactor='1'></PannerNode>
<AudioSource loop='true' url='sound/MEDIA_PARTS/EFFECTS_L.mp3' pitch='1.0' pauseTime='-1'
resumeTime='-1' startTime='1' />
</AudioSound>
```

Εικόνα 21- Κώδικας σύνδεσης του ήχου στην σκηνή X3DOM

```
<Transform DEF='top_left_gr' translation='251.190063 0.000061 0.000000' rotation=
'-1.000000 -0.000000 0.000000 -3.141593' scale='-1.000000 -1.000000 -1.000000' >
  <Transform DEF='top_left1' translation='-0.000122 0.000000 665.991211' rotation=
  '1.000000 -0.000000 -0.000000 -0.022673' >
    <Shape>
```

Εικόνα 22- Transform group ενός ηχείου της σκηνής

Στις δυο εικόνες που φαίνονται παραπάνω βλέπουμε ότι το γκρουπ Transform του AudioSound κάνει USE το 'top\_left1' και είναι το id που έχει προσδιοριστεί (DEF) στο γκρουπ Transform που περιλαμβάνει το ηχείο στη σκηνή.

Ο PannerNode χρησιμοποιείται για να δημιουργήσει την αίσθηση χωροθέτησης του ήχου, περισσότερες πληροφορίες και ανάλυση των χαρακτηριστικών του PannerNode γίνεται στην ενότητα **3.1.2.2 Διεπαφή PannerNode**.

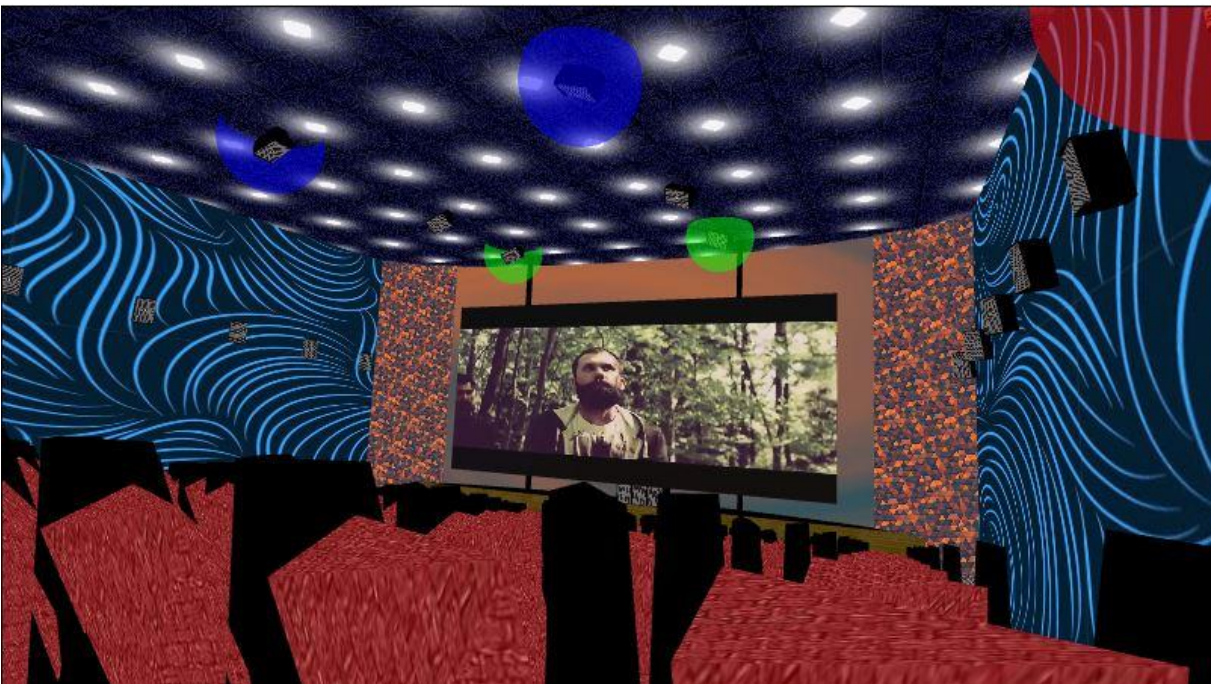
Το AudioSource περιέχει πληροφορίες για τον τρόπο αναπαραγωγής και την προέλευση του ήχου που θα αναπαραχθεί, περισσότερες πληροφορίες και ανάλυση των χαρακτηριστικών του AudioSource γίνεται στην ενότητα **4.1.3 Ενισχυμένοι κόμβοι X3D**.

Πέρα από τη βασική περιήγηση που προσφέρεται στην σκηνή του X3DOM, έχουμε προσθέσει ένα επιπλέον χαρακτηριστικό. Τη δυνατότητα να επιλέγει ο χρήστης συγκεκριμένη θέση στη σκηνή, κάνοντας κλικ πάνω σε κάποια πολυθρόνα του κινηματογράφου. Με τον τρόπο αυτό ο χρήστης μπορεί να παρατηρήσει την οπτική της θέσης προς την οθόνη ενώ παράλληλα παρατηρεί και την ακουστική σε αυτό το σημείο της σκηνής. Η δυνατότητα αυτή επιτεύχθηκε με τη χρήση ενός σεναρίου (JavaScript), το οποίο μεταφέρει τη θέση του viewpoint στη θέση της πολυθρόνας.

Τέλος, παρουσιάζουμε κάποιες εικόνες από την περιήγηση της σκηνής του κινηματογράφου.



Εικόνα 23- Οπτική γωνία πολυθρόνας κεντρικού τμήματος



Εικόνα 24- Οπτική γωνία πολυθρόνας δεξιού τμήματος



Εικόνα 25- Γενική οπτική γωνία της σκηνής του κινηματογράφου

## 5 Αποτελέσματα

### 5.1 Συμπεράσματα

Ακόμα κι αν τα πλεονεκτήματα του X3DOM σε σύγκριση με το X3D, δεν απαιτούν πρόσθετα, το X3DOM δεν θα μπορούσε να χειριστεί επαρκώς τον χωρικό ήχο. Αυτό ήταν ένα σημαντικό μειονέκτημα για το σχεδιασμό των ρεαλιστικών και διαδραστικών 3D σκηνών. Η πρότασή μας είναι ένας επαρκής τρόπος για να λυθεί αυτό το πρόβλημα και προσθέτει τον χωρικό ήχο στο X3DOM, διασφαλίζοντας ότι η ποιότητα των 3D σκηνών θα αυξηθεί. Ουσιαστικά, η εφαρμογή των ήδη υφιστάμενων κόμβων ήχου του X3D ήταν ανεπαρκής. Για το λόγο αυτό, συμπεριλάβαμε τη δομή των κόμβων του Web Audio API και τους προσαρμόσαμε στο πλαίσιο του X3DOM. Τέλος, με βάση τα παραδείγματα-πειράματα, μπορεί να εξαχθεί το συμπέρασμα ότι η διαδραστική σκηνή Web3D μπορεί να απαρτίζεται με τη χρήση των νέων εγγεγραμμένων κόμβων και τα αποτελέσματα επιβεβαιώνουν τη μεθοδολογία μας.

### 5.2 Μελλοντική εργασία και επεκτάσεις

Με την ολοκλήρωση αυτής της πτυχιακής και την επιτυχή ενσωμάτωση του χωρικού ήχου στο X3DOM, είναι πολλές οι δυνατότητες και οι προοπτικές εξέλιξης για περαιτέρω ανάπτυξη του ήχου σε διαδραστικές εφαρμογές και στο διαδίκτυο.

Σε μελλοντική έρευνα μπορεί να γίνει επέκταση σε περισσότερες δυνατότητες του ήχου σε συνδυασμό με την φυσική για τη δημιουργία νέων εφέ όπως για παράδειγμα η αντανάκλαση του ήχου για πιο ρεαλιστικά αποτελέσματα.

Το κεφάλαιο του ήχου στο X3DOM βρίσκεται σε πολύ αρχικό στάδιο, αυτό είναι θετικό καθώς υπάρχουν ενθαρρυντικές προοπτικές εξέλιξής του. Με αυτόν τον τρόπο στοχεύουμε στη δημιουργία νέων και πιο ρεαλιστικών 3D σκηνών και εφαρμογών, ανεβάζοντας έτσι το επίπεδο εμπειρίας στην περιήγηση εικονικών κόσμων.

## Βιβλιογραφία

- [1] Raggett, D. (1995). Extending WWW to support platform independent virtual reality. Technical Report.
- [2] KHROSOS GROUP. Retrieved from <https://www.khronos.org/news/press/khronos-releases-final-webgl-1.0-specification> [accessed May 2015].
- [3] Eicke, T. N., Jung, Y., & Kuijper, A. (2015). Stable dynamic webshadows in the X3DOM framework. Expert Systems with Applications. Retrieved from <http://dx.doi.org/10.1016/j.eswa.2014.11.059>
- [4] Daly, L., & Brutzman, D. (2007). X3D: Extensible 3D Graphics Standard. 24, pp. 130 - 135. IEEE. doi:10.1109/MSP.2007.905889
- [5] McIntosh, P., Hamilton, M., & Schyndel, R. v. (2005). X3D-UML: Enabling Advanced UML Visualisation Through X3D. Association for Computing Machinery, Inc.
- [6] Behr, J., Jung, Y., Keil, J., Drevensek, T., Zoellner, M., Eschler, P., & Fellner, D. (2010). A Scalable Architecture for the HTML5/ X3D Integration Model X3DOM. Web3D '10 Proceedings of the 15th International Conference on Web 3D Technology. doi:10.1145/1836049.1836077
- [7] Garbe, K., & Herbst, I. (2008). Extending X3D with Perceptual Auditory Properties. Virtual Reality Conference, 2008. VR '08. IEEE. doi:10.1109/VR.2008.4480787
- [8] Ding, H., Schwarz, D., Jacquemin, C., & Cahen, R. (2011). Spatial audio: graphic modeling for X3D. Web3D '11 Proceedings of the 16th International Conference on 3D Web Technology. doi:10.1145/2010425.2010427
- [9] W3C. (2013). Web Audio API. Retrieved from <http://www.w3.org/TR/webaudio/> [accessed May 2015].
- [10] Wyse, L. (2014). Interactive Audio Web Development Workflow. ACM. doi:10.1145/2647868.2655064
- [11] Wyse, L., & Subramanian, S. (2013). The viability of the web browser as a computer music platform (Vol. 37). (C. M. Journal, Ed.)
- [12] Dibra, D., Otero, N., & Pettersson, O. (2014). Real-Time Interactive Visualization Aiding Pronunciation of English as a Second Language. IEEE. doi:10.1109/ICALT.2014.131
- [13] Extensible 3D (X3D) ISO/IEC 19775-1:2008. Retrieved from <http://www.web3d.org/documents/specifications/19775-1/V3.2/Part01/Architecture.html> [accessed May 2015]
- [14] Ding, H., Schwarz, D., Jacquemin, C., Cahen, R. (2011). Spatial audio: graphic modeling for X3D. Web3D '11 Proceedings of the 16th International Conference on 3D Web Technology
- [15] Parisi, T. (2014). Programming 3D Applications with HTML5 and WebGL. O'Reilly Media, Inc
- [16] Spala, P., Malamos, A. G., Doulamis, A., & Mamakis, G. (2011). Extending MPEG-7 for efficient annotation of complex web 3D scenes (Vol. 59). Multimedia Tools and Applications, Springer US. doi:10.1007/s11042-011-0790-5
- [17] Stewart, J. A., Dumoulin, S. J., & Noël, S. (2006). X3D Conformance Testing Factors in Creating Aviable Test Suite. Electrical and Computer Engineering, 2006. CCECE '06. doi:10.1109/CCECE.2006.277506
- [18] Bouras, C., Panagopoulos, A., & Tsiatsos, T. (2005). Advances in X3D multi-user virtual environments. Multimedia, Seventh IEEE International Symposium. doi:10.1109/ISM.2005.28

- [19] Behr, J., Eschler, P., Jung, Y., & Zollner, M. (2009). X3DOM: a DOMbased HTML5/X3D integration model. Web3D '09 Proceedings of the 14th International Conference on 3D Web Technology. doi:10.1145/1559764.1559784
- [20] X3DOM. Retrieved from <http://www.x3dom.org/> [accessed May 2015]
- [21] Baglivo, A., Ponti, F. D., Luca, D. D., & Fanini, B. (2013). X3D/X3DOM, Blender Game Engine and OSG4WEB: open source visualisation for cultural heritage environments. Digital Heritage International Congress (DigitalHeritage).
- [22] Stamoulias, A., Malamos, A. G., Zampoglou, M., & Brutzman, D. (2014). Enhancing X3DOM declarative 3D with rigid body physics support. Proceedings of the Nineteenth International ACM Conference on 3D Web Technologies. ACM.
- [23] Mora-Lumbreras, M. A., Flores-Pulido, L., González-Contreras, B. M., & Portilla, A. (2012). Incorporating 3D Sound in Different Virtual Worlds. Web Congress (LA-WEB). doi:10.1109/LA-WEB.2012.17
- [24] Kapetanakis, K., Panagiotakis, S., Malamos, A. G., & Zampoglou, M. (2014). Adaptive Video Streaming on top of Web3D A bridging technology between X3DOM and MPEG-DASH. Telecommunications and Multimedia (TEMU), 2014 International Conference, IEEE. doi:10.1109/TEMU.2014.6917765
- [25] Wu, J.-R., Duh, C.-D., Ouhyoung, M., & Wu, J.-T. (1997). Head Motion and Latency Compensation on. Lausanne Switzerland: ACM VRST '97.
- [26] Lee, H. C., Kim, H. B., Lee, M. J., Kim, P. M., Suh, S. W., & Kim, K. H. (1998). Development of 3D sound generation system for multimedia application. Computer Human Interaction. IEEE.
- [27] Cowan, B., & Kapralos, B. (2013). Spatial sound rendering for dynamic virtual environments. Digital Signal Processing (DSP). IEEE.
- [28] Laitinen, M. V., Pihlajamäki, T., Erkut, C., & Pulkki, V. (2012). Parametric time-frequency representation of spatial sound in virtual worlds. ACM Transactions on Applied Perception (TAP). doi:10.1145/2207216.2207219
- [29] Smus, B. (2013). Web Audio API. O'Reilly Media, Inc
- [30] Spuy, R. v. (2015). Sound with the Web Audio API. In Advanced Game Design with HTML5 and JavaScript. IEEE.
- [31] Smus, B. Retrieved from Developing Game Audio with the Web Audio API: <http://www.html5rocks.com/en/tutorials/webaudio/games/> [accessed May 2015]
- [32] Roberts, C., Wakefield, G., & Wright, M. (2013). The Web Browser As Synthesizer And Interface. NIME.
- [33] Walker, W., & Belet, B. (2015). Birds of a Feather (Les Oiseaux de Même Plumage): Dynamic Soundscapes using Real-time Manipulation of Locally Relevant Birdsongs.
- [34] openAL. Retrieved from <http://openal.org/> [accessed May 2015]
- [35] Seo, B., Htoon, M. M., Zimmermann, R., & Wang, C.-D. (2010). Spatializer: A Web-based Positional Audio Toolkit. ACE. ACM.
- [36] Dirksen, J., Learning Three.js – the JavaScript 3D Library for WebGL, (2015)
- [37] Three.js Retrieved from <http://threejs.org> [accessed May 2015]
- [38] GidHub, webaudiox, Retrieved from
- [39] <https://github.com/jeromeetienne/webaudiox> [accessed May 2015]

- [40] GidHub, howler.js. Retrieved from <https://github.com/goldfire/howler.js> [accessed May 2015]
- [41] GidHub, pedalboard.js. Retrieved from
- [42] <http://dashersw.github.io/pedalboard.js/> [accessed May 2015]
- [43] GidHub, wad, Retrieved from <https://github.com/rserota/wad> [accessed May 2015]
- [44] GidHub, fifer, Retrieved from <https://github.com/f5io/fifer-js> [accessed May 2015]
- [45] DevBattles. Learn Web Audio API. Retrieved from <http://www.devbattles.com/sand/post-69-Learn-Web-Audio-API> [accessed May 2015]
- [46] Toyosak, A., Flahive, L., & Diaz, J. (2015). Music Story 2: a Motion-Graphic-based Music Video Creator.
- [47] Mahadevan, A., Freeman, J., & Magerko, B. (2015). EarSketch: Teaching computational music remixing in an online Web Audio based learning environment. Web Audio Conference.
- [48] Paradis, M., Clarke, R. G., & Melchior, F. (2015). VenueExplorer, Object- Based Interactive Audio for Live Events. WAC. Paris
- [49] Wyse, L. (2015). Spatially Distributed Sound Computing and Rendering Using the Web Audio Platform. 1st Web Audio Conference. Paris.
- [50] Pendharkar, C., Bäck, P., & Lonce, W. (2015). Adventures in scheduling, buffers and parameters: Porting a dynamic audio engine to Web Audio.
- [51] Schnell, N., Saiz, V., Barkati, K., & Goldszmidt, S. (2015). Of Time Engines and Masters An API for Scheduling and Synchronizing the Generation and Playback of Event Sequences and Media Streams for the Web Audio API. WAC. Paris.
- [52] Clark, C., & Tindale, A. (2014). Flocking: A Framework for Declarative Music-Making on the Web. ICMC|SMC. Athens.
- [53] Zampoglou, M., Spala, P., Kontakis, K., Malamos, A. G., & Ware, J. A. (2013). Direct Mapping of X3D Scenes to MPEG-7 Descriptions. Web3D '13, ACM. doi:10.1145/2466533.2466540
- [54] Pohja, M. (2003). X3D and VRML Sound Components.
- [55] Extensible 3D (X3D) Sound component. Retrieved from <http://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/sound.html> [accessed May 2015]
- [56] Ahmad, L., Boukerche, A., Hamidi, A., Shadid, A., Pazzi, R. (2008). Webbased e-learning in 3D large scale distributed interactive simulations using HLA/RTI. Parallel and Distributed Processing, IEEE
- [57] Repplinger, M., Loffler, A., Schug, B., Slusallek, P. (2009). Proceedings of the 14th International Conference on 3D Web Technology, ACM
- [58] Browsers support Web Audio API. Retrieved from <http://caniuse.com/#feat=audio-api> [accessed May 2015]
- [59] HTML5 (HyperText Markup Language). Retrieved from <https://en.wikipedia.org/wiki/HTML5>
- [60] JavaScript HTML DOM. Retrieved from [http://www.w3schools.com/js/js\\_htmlDOM.asp](http://www.w3schools.com/js/js_htmlDOM.asp)
- [61] JavaScript. Retrieved from <https://en.wikipedia.org/wiki/JavaScript>
- [62] Web Audio API Specification <https://webaudio.github.io/web-audio-api/>
- [63] Stamoulias, A., Lakka, E., & Malamos, A. G. (2015). "Wrapping" X3DOM around Web Audio API. International Journal of Artificial Intelligence and Interactive Multimedia, 3(Regular Issue)