



# **Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**

**Σχολή Τεχνολογικών Εφαρμογών**

**Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων**



## **Πτυχιακή Εργασία**

**Τίτλος : OpenAL Εφαρμογή σε Java για την παραγωγή  
αρχείων ήχου σε 6 κανάλια**

**Δημήτρης Παπαδάκης 1075**

**Επιβλέπων Καθηγητής : Αθανάσιος Μαλάμος**

**Ημερομηνία Παρουσίασης : Νοέμβριος 2012**

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστήσω τον εισηγητή κύριο Μαλάμο Αθανάσιο και το βοηθό του Καπετανάκη Κωνσταντίνο για τη συμμετοχή τους στην ολοκλήρωση της πτυχιακής εργασίας.

## ΣΥΝΟΨΗ

Σκοπός της εργασίας είναι να υλοποιήσουμε το api του x3d. Θα φτιάξουμε μια εφαρμογή στην οποία ο χρήστης φορτώνει ένα αρχείο ήχου wav και το αναπαράγει. Ζητούμενο είναι ο χρήστης να μπορεί να επιλέξει το ηχείο το οποίο θα αναπαράγει τον ήχο. Η εφαρμογή παρέχει υποστήριξη πολυκάναλου, έξι καναλιών, surround συστήματος ήχου χρησιμοποιώντας τεχνολογία OpenAL.

## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

### **1 Εισαγωγή στη Java**

- 1.1 *Ιστορία*
- 1.2 *Αρχές της Java*
- 1.3 *Πλατφόρμα λογισμικού και Java Virtual Machine*
- 1.4 *Σύνταξη*
- 1.5 *Βιβλιοθήκες (Class Libraries)*

### **2 Εισαγωγή στο NetBeans**

- 2.1 *Πλατφόρμα NetBeans*

### **3 Surround Sound**

- 3.1 *Πεδία Εφαρμογή Surround*
- 3.2 *Τύποι Μέσων και Τεχνολογιών*
- 3.3 *Δημιουργία ήχου Surround*
- 3.4 *Χαρτογράφηση καναλιών στα ηχεία*
- 3.5 *Διαχείριση μπάσου – Low Frequency Effects Channel*
- 3.6 *Ambisonics – PanAmbio*

### **4 3D Audio**

- 4.1 *Διαπλάτνση Stereo*
- 4.2 *Complete 3D Positional Audio*
- 4.3 *Προτεραιότητα Ήχου*
- 4.4 *Εξασθένηση Ήχου και spatialization*
- 4.5 *Audio Clip*
- 4.6 *Sound Node*

### **5 OpenAL**

- 5.1 *Δομή και λειτουργικότητα του API*
- 5.2 *Μειονεκτήματα*
- 5.3 *Φορητότητα*

### **6 Η Εφαρμογή μας (Audio Player)**

- 6.1 *JFileChooser*
- 6.2 *Play, Stop Button*
- 6.3 *X, Y, Z Position*
- 6.4 *Gain – Pitch – Looping*
- 6.5 *Απαιτούμενες Βιβλιοθήκες*

### **Βιβλιογραφία**

## **ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ**

**Εικόνα 1 : Stereo Panning**

**Εικόνα 2 : Γεωμετρία Κόμβου Ήχου**

**Εικόνα 3 : Audio Player**

**Εικόνα 4 : Τυπώνουμε το αρχείο που φορτώσαμε**



## 1 ΕΙΣΑΓΩΓΗ ΣΤΗ JAVA

Η Java είναι μια γλώσσα προγραμματισμού που αναπτύχθηκε αρχικά από τον James Gosling στη Sun Microsystems (η οποία έκτοτε έχει συγχωνευθεί με την Oracle Corporation) και κυκλοφόρησε το 1995 ως βασική συνιστώσα της πλατφόρμας Java της Sun Microsystems. Η σύνταξη της γλώσσας προέρχεται κατά μεγάλο μέρος από C και C++, αλλά έχει ένα απλούστερο μοντέλο αντικειμένου και λιγότερες χαμηλού επιπέδου εγκαταστάσεις είτε από C ή C++. Οι εφαρμογές Java συνήθως συγκεντρώνονται σε bytecode (αρχείο class) που μπορεί να τρέξει σε οποιαδήποτε Java Virtual Machine (JVM), ανεξάρτητα από την αρχιτεκτονική του υπολογιστή. Η Java είναι μια γενικής χρήσης, βασισμένη σε κλάσεις, αντικειμενοστραφή γλώσσα που έχει σχεδιαστεί ειδικά για να έχει όσο το δυνατόν λιγότερες εξαρτήσεις στην υλοποίηση. Σκοπός του είναι να επιτρέψει στους προγραμματιστές εφαρμογών να "write once, run anywhere" (WORA), πράγμα που σημαίνει ότι κώδικας που τρέχει σε μια πλατφόρμα δεν χρειάζεται να επαναμεταγλωττιστεί για να τρέξει σε μια άλλη. Η Java για το 2012, είναι μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού σε χρήση, ιδιαίτερα για client-server εφαρμογές web, με 10 εκατομμύρια αναφερθέντες χρήστες.

Η αρχική υλοποίηση των Java compilers (μεταγλωτιστών Java), εικονικών μηχανών, και οι βιβλιοθήκες τάξεων (class libraries), έχουν αναπτυχθεί από τη Sun από το 1995. Από το Μάιο του 2007, σύμφωνα με τις προδιαγραφές της διαδικασίας κοινοτικής Java Community Process, η Sun επανέκδωσε τις περισσότερες τεχνολογίες Java υπό την GNU General Public License. Άλλοι έχουν επίσης αναπτύξει εναλλακτικές εφαρμογές των τεχνολογιών αυτών της Sun, όπως το GNU Compiler για Java και το GNU Classpath.

### 1.1 ΙΣΤΟΡΙΑ

Η Sun Microsystems κυκλοφόρησε την πρώτη δημόσια υλοποίηση ως Java 1.0 το 1995. Υποσχέθηκε "write once, run anywhere" (WORA), παρέχοντας χωρίς κόστος εκτελέσεις σε δημοφιλείς πλατφόρμες. Αρκετά ασφαλές και διαθέτοντας δυνατότητα ρύθμισης της ασφάλειας, επέτρεψε περιορισμούς δικτύου και πρόσβασης αρχείων. Σημαντικοί περιηγητές δικτύου ενσωμάτωσαν σύντομα τη δυνατότητα να τρέχουν Java applets σε ιστοσελίδες και η Java γρήγορα έγινε δημοφιλής. Με την έλευση της Java 2 (κυκλοφόρησε αρχικά ως J2SE 1.2 το Δεκέμβριο 1998-1999), νέες εκδόσεις είχαν πολλαπλές διαμορφώσεις κατασκευασμένες για διαφορετικούς τύπους πλατφόρμας. Για παράδειγμα, J2EE στοχευμένες εταιρικές εφαρμογές και η πολύ λιτή έκδοση J2ME εφαρμογών για κινητά τηλέφωνα (Mobile Java). Η J2SE όρισε τη Standard Edition. Το 2006, για λόγους μάρκετινγκ, η Sun μετονόμασε νέες εκδόσεις J2 όπως η Java EE, Java ME, και Java SE, αντίστοιχα.

Το 1997, η Sun Microsystems προσέγγισε το πρότυπο ISO / IEC JTC1 και αργότερα το Ecma International για την επισημοποίηση της Java, αλλά σύντομα αποσύρθηκε από τη διαδικασία. Η Java παραμένει ένα de facto πρότυπο, που ελέγχεται μέσω της διαδικασίας της Κοινότητας Java (Java Community Process). Κάποια στιγμή, η Sun έκανε το μεγαλύτερο μέρος των Java εφαρμογών που διατίθενται χωρίς χρέωση, παρά το ιδιοκτησιακό καθεστώς του λογισμικού τους. Η Sun παρήγαγε έσοδα από την Java, μέσω της πώλησης των αδειών για τα εξειδικευμένα προϊόντα, όπως το Java Enterprise System. Η Java διακρίνει το Software Development Kit της (SDK) από το Runtime Environment (JRE) (ένα υποσύνολο του SDK): Η κύρια διάκριση αναφέρεται στην έλλειψη του μεταγλωττιστή, βοηθητικών προγραμμάτων και αρχείων κεφαλίδας (header files) του JRE.

Στις 13 Νοεμβρίου 2006, η Sun κυκλοφόρησε ένα μεγάλο μέρος της Java ως ελεύθερο και ανοικτού κώδικα λογισμικό, υπό τους όρους της GNU General Public License (GPL). Στις 8 Μαΐου 2007, η Sun τερμάτισε τη διαδικασία, κάνοντας όλο τον πυρήνα του κώδικα της Java που διατίθενται στο πλαίσιο ελεύθερου λογισμικού / ανοικτού κώδικα όσον αφορά τη διανομή, εκτός από ένα μικρό τμήμα του κώδικα με τον οποίο η Sun δεν κατέχει τα πνευματικά δικαιώματα.

## 1.2 ΑΡΧΕΣ ΤΗΣ JAVA

Υπήρχαν πέντε πρωταρχικοί στόχοι στη δημιουργία της γλώσσας Java:

- Θα πρέπει να είναι "απλή, αντικειμενοστραφής και οικεία"
- Θα πρέπει να είναι "ισχυρή και ασφαλής"
- Θα πρέπει να είναι " ουδέτερης αρχιτεκτονικής και φορητή"
- Θα πρέπει να εκτελέσει με "υψηλή απόδοση"
- Θα πρέπει να είναι "ερμηνεύσιμη, δυναμική και threaded "

## 1.3 ΠΛΑΤΦΟΡΜΑ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ JAVA VIRTUAL MACHINE

Ένα από τα χαρακτηριστικά της Java είναι η φορητότητα (portability), πράγμα που σημαίνει ότι τα προγράμματα ηλεκτρονικών υπολογιστών, γραμμένα στη γλώσσα Java μπορούν να τρέχουν σε οποιαδήποτε παρόμοια πλατφόρμα υλικού / λειτουργικού συστήματος. Αυτό επιτυγχάνεται με τη μεταγλώττιση του κώδικα γλώσσας Java σε μια ενδιάμεση αναπαράσταση που ονομάζεται Java bytecode, αντί της απευθείας σε συγκεκριμένη πλατφόρμα κώδικα μηχανής. Οι οδηγίες Java bytecode είναι ανάλογες με κώδικα μηχανής, αλλά προορίζονται να ερμηνεύονται από μια εικονική μηχανή (VM), γραμμένες ειδικά για το υλικό υποδοχής. Οι τελικοί χρήστες χρησιμοποιούν συχνά το Java Runtime Environment (JRE) που έχουν εγκαταστήσει σε δική τους μηχανή για αυτόνομες εφαρμογές Java, ή σε ένα πρόγραμμα περιήγησης στο Web για βοηθητικές εφαρμογές Java (Java Applets).

Τυποποιημένες βιβλιοθήκες παρέχουν ένα γενικό τρόπο για την πρόσβαση χαρακτηριστικών ειδικών υποδοχής (host specific) , όπως γραφικά, threading, και δικτύωση.

Ένα σημαντικό όφελος από τη χρήση bytecode είναι η φορητότητα. Ωστόσο, η επιβάρυνση της ερμηνείας σημαίνει ότι τα προγράμματα που ερμηνεύονται σχεδόν πάντα τρέχουν πιο αργά από ότι τα προγράμματα που μεταγλωττίζονται σε αυτόχθονα εκτελέσιμα. Just-in-Time (JIT) μεταγλωττιστές εισήχθησαν από νωρίς οι οποίοι μεταγλωττίζουν bytecodes σε κώδικα μηχανής κατά τη διάρκεια της εκτέλεσης.

## 1.4 ΣΥΝΤΑΞΗ

Η σύνταξη της Java προέρχεται κυρίως από τη C ++. Σε αντίθεση με τη C ++, η οποία συνδυάζει τη σύνταξη για δομημένο, γενικό, και αντικειμενοστραφή προγραμματισμό, η Java χτίστηκε σχεδόν αποκλειστικά ως μία αντικειμενοστραφής γλώσσα. Όλος ο κώδικας είναι γραμμένος μέσα σε μια κατηγορία (class), και όλα είναι ένα αντικείμενο, με την εξαίρεση των πρωτόγονων τύπων δεδομένων (π.χ. ακέραιοι, αριθμοί κινητής υποδιαστολής, δυαδικές τιμές, και χαρακτήρες), οι οποίοι δεν είναι κλάσεις για λόγους απόδοσης.

Σε αντίθεση με τη C ++, η Java δεν υποστηρίζει υπερφόρτωση χειριστή (operator overloading) ή πολλαπλή κληρονομικότητα για τις κλάσεις. Αυτό απλοποιεί τη γλώσσα και βοηθάει στην πρόληψη πιθανών λαθών και στο σχεδιασμό.

## 1.5 BIBΛΙΟΘΗΚΕΣ (CLASS LIBRARIES)

Η Java Class Library είναι τα μεταγλωττισμένα bytecodes του πηγαίου κώδικα που αναπτύχθηκαν από τον υλοποιητή JRE για να υποστηρίξει την ανάπτυξη εφαρμογών σε Java.

Παραδείγματα αυτών των βιβλιοθηκών είναι:

- Οι βιβλιοθήκες πυρήνα (core libraries), οι οποίες περιλαμβάνουν:
  - Βιβλιοθήκες συλλογής (Collection libraries) που υλοποιούν δομές δεδομένων όπως λίστες, λεξικά, δέντρα, σύνολα, ουρές και ουρές διπλής απόλωσης, ή στοιβες
  - Βιβλιοθήκες Επεξεργασίας XML (Parsing, Μετασχηματισμός, επικύρωση)
  - Ασφάλεια
  - Βιβλιοθήκες διεθνοποίησης και εντοπισμού
- Οι βιβλιοθήκες ένταξης, οι οποίες επιτρέπουν στο συγγραφέα της εφαρμογής να επικοινωνούν με εξωτερικά συστήματα. Οι βιβλιοθήκες αυτές περιλαμβάνουν:
  - Τη Java Database Connectivity (JDBC) API για πρόσβαση σε βάσεις δεδομένων
  - Java Naming και Directory Interface (JNDI) για την αναζήτηση και ανακάλυψη
  - RMI και CORBA για την κατανεμημένη ανάπτυξη εφαρμογών
  - JMX για τη διαχείριση και τον έλεγχο εφαρμογών
- Βιβλιοθήκες διεπαφή χρήστη, οι οποίες περιλαμβάνουν:
  - Η (βαριά, ή μητρική) Abstract Window Toolkit (AWT), η οποία παρέχει GUI στοιχεία, τα μέσα για την τοποθέτηση των εν λόγω στοιχείων και τα μέσα για τη διαχείριση των γεγονότων από τα συστατικά
  - Οι (ελαφριές) Swing βιβλιοθήκες, οι οποίες είναι χτισμένες σε AWT, αλλά παρέχουν (μη μητρικές) εφαρμογές της AWT widgetry
  - APIs για σύλληψη ήχου, επεξεργασία και αναπαραγωγή
- Μια υλοποίηση της Java Virtual Machine που εξαρτάται από την πλατφόρμα, που είναι ο τρόπος με τον οποίο εκτελούνται τα bytecodes των βιβλιοθηκών Java και εφαρμογές τρίτων
- Πρόσθετα (Plugins), τα οποία επιτρέπουν applets να τρέξουν σε προγράμματα περιήγησης στο Web
- Java Web Start, η οποία επιτρέπει σε εφαρμογές Java να διανεμηθούν αποτελεσματικά στους τελικούς χρήστες μέσω του Internet
- Αδειοδότηση και τεκμηρίωση.





## 2 ΕΙΣΑΓΩΓΗ ΣΤΟ NETBEANS

Το NetBeans αναφέρεται τόσο σ' ένα πλαίσιο πλατφόρμας για Java desktop εφαρμογές όσο και σ' ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) για ανάπτυξη με Java, JavaScript, PHP, Python (δεν υποστηρίζεται πλέον μετά το NetBeans 7), groovy, C, C ++, Scala, Clojure και άλλα. Το NetBeans IDE 7,0 επίσημα πλέον δεν υποστηρίζει Ruby και Ruby on Rails, αλλά ένα plug-in επιτρέπει την ανάπτυξη.

Το NetBeans IDE είναι γραμμένο σε Java και μπορεί να τρέξει σε Windows, OS X, Linux, Solaris και σε άλλες πλατφόρμες που υποστηρίζουν ένα συμβατό JVM. Μια προϋπάρχουσα JVM ή ένα JDK δεν απαιτείται.

Η πλατφόρμα NetBeans επιτρέπει ανάπτυξη εφαρμογών από ένα σύνολο αρθρωτών εξαρτημάτων λογισμικού που ονομάζεται ενότητες (modules). Οι εφαρμογές που βασίζονται στην πλατφόρμα NetBeans (συμπεριλαμβανομένου του NetBeans IDE) μπορεί να επεκταθεί από τρίτους προγραμματιστές.

### 2.1 ΠΛΑΤΦΟΡΜΑ NETBEANS

Η πλατφόρμα NetBeans είναι ένα επαναχρησιμοποιήσιμο πλαίσιο εργασίας για την απλοποίηση της ανάπτυξης Java Swing desktop εφαρμογών. Το NetBeans IDE πακέτο για Java SE περιέχει ό, τι χρειάζεται για να αρχίσει η ανάπτυξη του NetBeans plugins και εφαρμογών στηριγμένες στην πλατφόρμα NetBeans. Κανένα επιπλέον SDK δεν απαιτείται.

Οι εφαρμογές μπορούν να εγκαταστήσουν ενότητες (modules) δυναμικά. Κάθε εφαρμογή μπορεί να περιλαμβάνει την ενότητα Κέντρο Ενημέρωσης (Update Center) για να επιτρέψει στους χρήστες της εφαρμογής να κατεβάσουν ψηφιακά υπογεγραμμένες αναβαθμίσεις και νέα χαρακτηριστικά απ' ευθείας στην εφαρμογή που εκτελείται. Επανεγκατάσταση μιας αναβάθμιση ή μιας νέα έκδοση δεν επιβάλλει στους χρήστες να κατεβάσουν ολόκληρη την εφαρμογή ξανά.

Η πλατφόρμα προσφέρει επαναχρησιμοποιήσιμες υπηρεσίες κοινές για desktop εφαρμογές, επιτρέποντας στους προγραμματιστές να επικεντρωθούν στην λογική ειδικά για την εφαρμογή τους. Μεταξύ των χαρακτηριστικών της πλατφόρμας είναι:

- Διαχείριση Διεπαφής Χρήστη (User Interface Management) (π.χ. μενού και γραμμές εργαλείων)
- Διαχείριση ρυθμίσεων χρήστη
- Διαχείριση αποθήκευσης (αποθήκευση και φόρτωση κάθε είδους δεδομένων)
- Διαχείριση Παραθύρων (Windows Management)
- Οδηγός πλαισίου εργασίας (υποστηρίζει βήμα-προς-βήμα διαλόγους)
- Εικονική Βιβλιοθήκη NetBeans (NetBeans Visual Library)
- Ολοκληρωμένα εργαλεία ανάπτυξης

Το NetBeans IDE είναι ένα δωρεάν, ανοιχτού κώδικα, cross-platform (ανεξάρτητης πλατφόρμας) IDE με ενσωματωμένη υποστήριξη για γλώσσα προγραμματισμού Java.

Surround ήχος είναι μια τεχνική για τον εμπλουτισμό της ποιότητας αναπαραγωγή ήχου μιας πηγής ήχου με επιπλέον κανάλια ήχου από ηχεία που περιβάλλουν τον ακροατή (surround κανάλια), παρέχοντας ήχο από μια ακτίνα 360 ° στο οριζόντιο επίπεδο (2D) σε αντίθεση με το "κανάλια στην οθόνη" (κέντρο, [μπροστά] αριστερά, και [μπροστά] δεξιά) που προέρχονται μόνο από το εμπρός τόξο του ακροατή.

Ο ήχος Surround χαρακτηρίζεται από μια θέση ακροατή ή γλυκό σημείο όπου τα ηχητικά εφέ λειτουργούν καλύτερα, και παρουσιάζει μια σταθερή ή προς τα εμπρός προοπτική του ηχητικού πεδίου στον ακροατή σε αυτή την τοποθεσία. Η τεχνική αυτή βελτιώνει την αντίληψη κατανομής του ήχου στο χώρο με την αξιοποίηση του εντοπισμού του ήχου: Η ικανότητα του ακροατή να εντοπίσει τη θέση ή την προέλευση ενός ανιχνεύσιμου ήχου σε κατεύθυνση και απόσταση. Τυπικά, αυτό επιτυγχάνεται με τη χρήση πολλαπλών διακριτών κανάλια ήχου που οδηγούνται σε μια συστοιχία μεγαφώνων.

Υπάρχουν διάφορες τεχνικές και μορφές με βάση τον ήχο surround, οι οποίες ποικίλλουν στην αναπαραγωγή και την καταγραφή των μεθόδων όπως και στον αριθμό και τη θέση των επιπλέον καναλιών.

### 3.1 ΠΕΔΙΑ ΕΦΑΡΜΟΓΗΣ SURROUND

Αν και κινηματογράφο και soundtracks αντιπροσωπεύουν τις κύριες χρήσεις των τεχνικών surround, το πεδίο εφαρμογής της είναι ευρύτερο από αυτό όσο ο surround ήχο επιτρέπει για τη δημιουργία ενός περιβάλλοντος ήχου για όλους τους σκοπούς. Πολυκαναλικές τεχνικές ήχου μπορούν να χρησιμοποιηθούν για την αναπαραγωγή περιεχομένου τόσο διαφορετικά όπως μουσική, ομιλία, φυσικές ή συνθετικοί ήχοι για τον κινηματογράφο, την τηλεόραση, μετάδοση, ή υπολογιστές. Όσον αφορά το περιεχόμενο μουσικής για παράδειγμα, μια ζωντανή παράσταση μπορεί να χρησιμοποιεί πολυκαναλικές τεχνικές στο πλαίσιο μιας υπαίθριας συναυλίας, ενός μουσικού θεάτρου ή για αναμετάδοση: για μια ταινία συγκεκριμένες τεχνικές υιοθετούνται για τον κινηματογράφο, ή για σπίτι (π.χ. συστήματα home cinema). Ο χώρος της αφήγησης είναι επίσης ένα περιεχόμενο που μπορεί να ενισχυθεί μέσω πολυκαναλικών τεχνικών. Αυτό ισχύει κυρίως για αφηγήσεις του κινηματογράφου, για παράδειγμα, η ομιλία των χαρακτήρων της ταινίας, αλλά μπορεί επίσης να εφαρμοστεί σε έργα για το θέατρο, σε ένα συνέδριο, ή να ενσωματώσει τη φωνή με βάση τις παρατηρήσεις σε ένα αρχαιολογικό χώρο ή μνημείο. Για παράδειγμα, μια έκθεση μπορεί να ενισχυθεί με τον τοπικό περιβάλλον ήχο του θορύβου του νερού, των πουλιών, του τρένου ή του μηχανήματος. Τοπικοί φυσικοί ήχοι μπορούν επίσης να χρησιμοποιηθούν σε εκπαιδευτικές εφαρμογές. Άλλα πεδία εφαρμογής περιλαμβάνουν κονσόλες βιντεοπαιχνιδιών, προσωπικούς υπολογιστές και άλλες πλατφόρμες. Σε τέτοιες εφαρμογές, το περιεχόμενο θα είναι τυπικά συνθετικός θόρυβος, που παράγεται από τη συσκευή υπολογιστή σε αλληλεπίδραση με το χρήστη. Σημαντική δουλειά έχει γίνει επίσης με ήχο surround για αυξημένη επίγνωση της κατάστασης σε στρατιωτικές και δημόσιες ασφάλειας εφαρμογές.

### 3.2 ΤΥΠΟΙ ΜΕΣΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΩΝ

Τα εμπορικά μέσα surround ήχου περιλαμβάνουν βιντεοκασέτες, DVD και μεταδόσεις HDTV κωδικοποιούνται ως συμπιεσμένο Dolby Digital και DTS, και χωρίς απώλειες ήχου όπως DTS HD Master Audio και Dolby TrueHD σε Blu-ray Disc και HD DVD, τα οποία είναι πανομοιότυπα με

το master studio. Άλλες εμπορικές μορφές περιλαμβάνουν την ανταγωνιστική DVD-Audio (DVD-A) και Super Audio CD (SACD) μορφές, και MP3 Surround. Cinema 5.1 surround μοντέλο περιλαμβάνει Dolby Digital και DTS. Το Sony Dynamic Digital Sound (SDDS) είναι μια 8-κάναλη διαμόρφωση σινεμά η οποία διαθέτει 5 ανεξάρτητα κανάλια ήχου σε όλη την πρόσοψη με δύο ανεξάρτητα κανάλια surround, και ένα χαμηλής συχνότητας κανάλι εφέ. Η παραδοσιακή 7.1 surround διαμόρφωση εισάγει δύο επιπλέον πίσω ηχεία στο συμβατική 5,1 ρύθμιση, για ένα σύνολο τεσσάρων καναλιών surround και τρία μπροστινά κανάλια, για να δημιουργήσετε ένα πιο ευρυγώνιο 360 ° ηχητικό πεδίο.

Οι περισσότερες ηχογραφήσεις surround που δημιουργήθηκαν από τις εταιρείες παραγωγής ταινιών ή παραγωγούς βιντεοπαιχνιδιών: Ωστόσο, ορισμένες βιντεοκάμερες της αγοράς έχουν τέτοια δυνατότητα, είτε ενσωματωμένο ή διατίθενται χωριστά. Τεχνολογίες Surround ήχου μπορούν επίσης να χρησιμοποιηθούν για τη μουσική επιτρέποντας την ανάπτυξη νέων μεθόδων της καλλιτεχνικής έκφρασης. Μετά την αποτυχία του τετραφωνικού ήχου στη δεκαετία του 1970, πολυκάναλη μουσική έχει σιγά-σιγά έχουν εκ νέου επανεισαχθεί από το 1999 με τη βοήθεια του SACD και DVD-Audio. Μερικοί δέκτες AV, στερεοφωνικά συστήματα και κάρτες ήχου ηλεκτρονικών υπολογιστών περιλαμβάνουν ενσωματωμένους ψηφιακούς επεξεργαστές σήματος και / ή επεξεργαστές ψηφιακού ήχου για την προσομοίωση ήχο surround από μια στερεοφωνική πηγή.

Το 1967, το ροκ συγκρότημα των Pink Floyd πραγματοποίησε για πρώτη φορά συναυλία surround ήχου στο " Games for May ", μια πολυτελή υπόθεση στο Queen Elizabeth Hall του Λονδίνου, όπου η μπάντα έκανε το ντεμπούτο του αυτοσχέδιου τετραφωνικού συστήματος ηχείων.

### 3.3 ΔΗΜΙΟΥΡΓΙΑ ΗΧΟΥ SURROUND

Ο surround ήχος δημιουργείται με διάφορους τρόπους. Η πρώτη και απλούστερη μέθοδος είναι η χρήση μιας surround τεχνικής μικροφώνου, ηχογράφησης ήχου, ή / και μείξη σε surround ήχο για αναπαραγωγή σε ένα σύστημα ήχου με τα ηχεία που περιβάλλουν τον ακροατή για να παίξει τον ήχο από διαφορετικές κατευθύνσεις. Μια δεύτερη προσέγγιση είναι η επεξεργασία του ήχου με ψυχοακουστικές μεθόδους εντοπισμού ήχου για να προσομοιώσει ένα διδιάστατο (2-D) πεδίο ήχου με ακουστικά. Μια τρίτη προσέγγιση, με βάση την αρχή Huygens', επιχειρεί την ανασύσταση των εγγεγραμμένων μετώπων ηχητικού πεδίου κύματος μέσα στο χώρο ακρόασης: μία μορφή « ηχητικού ολογράμματος ». Μια μορφή, wave field synthesis (WFS), παράγει ένα ηχητικό πεδίο με ένα ίσο πεδίο λάθους σε όλη την περιοχή. Εμπορικά WFS συστήματα, που κυκλοφορούν σήμερα από τις εταιρείες sonic emotion και Isoono, απαιτούν πολλά ηχεία και σημαντική υπολογιστική ισχύ.

Η μορφή Ambisonics, επίσης, με βάση την αρχή Huygens', δίνει μια ακριβή ανακατασκευή του ήχου στο κεντρικό σημείο: Λιγότερο ακριβή μακριά από το κέντρο. Υπάρχουν πολλά δωρεάν και εμπορικά λογισμικά διαθέσιμα για Ambisonics, που κυριαρχεί στο μεγαλύτερο μέρος της καταναλωτικής αγοράς, ιδιαίτερα μουσικοί με τη χρήση ηλεκτρονικής μουσικής και υπολογιστών. Επιπλέον, Ambisonics προϊόντα είναι το πρότυπο σε υλικό surround ήχου που πωλείται από τη Meridian Audio, Ltd. Στην απλούστερη μορφή του, Ambisonics καταναλώνει ελάχιστους πόρους, ωστόσο αυτό δεν είναι αλήθεια για τις πρόσφατες εξελίξεις, όπως η Near Field Compensated Higher Order Ambisonics. Μερικά χρόνια πριν δείχθηκε ότι, στο όριο, WFS και Ambisonics συγκλίνουν.

Τέλος, ήχος surround, μπορεί να επιτευχθεί στο επίπεδο mastering, από στερεοφωνικές πηγές, όπως με Penteo, η οποία χρησιμοποιεί την ανάλυση Ψηφιακής Επεξεργασίας Σήματος μιας στερεοφωνικής εγγραφής για να αναλύσει μεμονωμένους ήχους σε συστατικές θέσεις πανόραμα, και μετά τις τοποθετεί ανάλογα, σ' ένα πεντακάναλο πεδίο. Υπάρχουν, ωστόσο, περισσότεροι τρόποι για να δημιουργήσετε surround από στερεοφωνικό, για παράδειγμα, με τις ρουτίνες βασισμένες στο QS και SQ Quad, όπου τα όργανα ήταν στο στούντιο χωρισμένα σε 4 ηχεία. Αυτός ο τρόπος δημιουργίας surround με ρουτίνες λογισμικού συνήθως αναφέρεται ως "upmixing".

### 3.4 ΧΑΡΤΟΓΡΑΦΗΣΗ ΚΑΝΑΛΙΩΝ ΣΤΑ ΗΧΕΙΑ

Στις περισσότερες περιπτώσεις, τα συστήματα ήχου surround βασίζονται στη χαρτογράφηση του κάθε καναλιού προέλευσης προς το ηχείο τους. Matrix συστήματα ανακτούν τον αριθμό και το περιεχόμενο των καναλιών προέλευσης και τα εφαρμόζουν στα αντίστοιχα ηχεία τους. Με διακριτό περιβάλλοντα ήχο, το μέσο μετάδοσης επιτρέπει (τουλάχιστον) τον ίδιο αριθμό σταθμών προέλευσης και προορισμού: Ωστόσο, ένα-προς-ένα, κανάλι-προς-ομιλητή χαρτογράφηση δεν είναι ο μόνος τρόπος για τη μετάδοση σημάτων ήχου surround.

Το μεταδιδόμενο σήμα μπορεί να κωδικοποιεί την πληροφορία (που ορίζει το αρχικό πεδίο ήχου), σε μεγαλύτερο ή μικρότερο βαθμό: Η surround πληροφορία ήχου καθίσταται για αναπαραγωγή από έναν αποκωδικοποιητή που παράγει τον αριθμό και τη διάταξη των καναλιών για τον αριθμό των διαθέσιμων ηχείων για αναπαραγωγή - μία καθιστά ένα ηχητικό πεδίο που παράγεται από ένα σύνολο ηχείων, κατά τρόπο ανάλογο με απόδοση σε γραφικά ηλεκτρονικών υπολογιστών. Αυτή η κωδικοποίηση "ανεξάρτητης συσκευής αναπαραγωγής" είναι ανάλογη με τον κωδικοποίηση και αποκωδικοποίηση του Adobe PostScript αρχείου, όπου το αρχείο περιγράφει την σελίδα, και αποδίδεται ανά την ικανότητα ανάλυσης της συσκευής εξόδου. Η Ambisonics και WFS συστήματα χρησιμοποιούν απόδοση ήχου (audio rendering): Η Meridian Lossless Packing περιέχει στοιχεία αυτής της δυνατότητας.

### 3.5 ΔΙΑΧΕΙΡΙΣΗ ΜΠΑΣΟΥ – LOW FREQUENCY EFFECTS CHANNEL

Τα surround συστήματα αναπαραγωγής μπορούν να κάνουν χρήση διαχείρισης του μπάσου, η θεμελιώδης αρχή της οποίας είναι ότι το περιεχόμενο του μπάσου στο εισερχόμενο σήμα, ανεξάρτητα από το κανάλι, θα πρέπει να απευθύνονται μόνο σε ηχεία ικανά να το χειρίζονται, είτε αυτά είναι τα κύρια ηχεία του συστήματος ή ένα ή περισσότερα είδη ηχείων χαμηλών συχνοτήτων που ονομάζονται subwoofers.

Υπάρχει μια διαφορά πριν και μετά το σύστημα διαχείρισης μπάσων. Πριν από το σύστημα διαχείρισης μπάσου υπάρχει ένα εφέ χαμηλής συχνότητας (Low Frequency Effects LFE) κανάλι. Μετά από το σύστημα διαχείρισης μπάσου υπάρχει ένα σήμα subwoofer. Μια κοινή παρανόηση είναι η πεποίθηση ότι το κανάλι LFE είναι το "κανάλι subwoofer". Το σύστημα διαχείρισης μπάσων μπορεί να κατευθύνει μπάσο σε ένα ή περισσότερα subwoofer (εάν υπάρχουν) από οποιοδήποτε κανάλι, όχι μόνο από το κανάλι LFE. Επίσης, αν δεν υπάρχει ηχείο subwoofer τότε το σύστημα διαχείρισης μπάσων μπορεί να κατευθύνει το κανάλι LFE σε ένα ή περισσότερα από τα κύρια ηχεία.

#### Low Frequency Effects (LFE) κανάλι

Επειδή το κανάλι χαμηλής συχνότητας εφέ απαιτεί μόνο ένα κλάσμα του εύρους ζώνης των άλλων καναλιών ήχου, αυτό αναφέρεται ως "0,1" κανάλι: Για παράδειγμα "5.1" ή "7.1".

Το LFE είναι πηγή κάποιας σύγχυσης στον ήχο surround. Το κανάλι LFE αναπτύχθηκε αρχικά για τη μεταφορά εξαιρετικά χαμηλών "sub-bass" κινηματογραφικών εφέ ήχου (με τα εμπορικά subwoofer μερικές φορές να πηγαίνουν κάτω έως 30 Hz, π.χ., το δυνατό βουητό της βροντής ή εκρήξεις) στο δικό τους κανάλι. Αυτό επέτρεπε στα θέατρα να ελέγξουν την ένταση αυτών των αποτελεσμάτων και να ταιριάζουν στο ακουστικό περιβάλλον του συγκεκριμένου κινηματογράφου

και στο ηχητικό σύστημα αναπαραγωγής. Ο ανεξάρτητος έλεγχος των sub-bass μείωσε επίσης το πρόβλημα της παραμόρφωσης ενδοδιαμόρφωσης (intermodulation distortion) στην αναλογικής ταινίας αναπαραγωγή ήχου.

Στην αρχική υλοποίηση του κινηματογράφου, το LFE ήταν ένα χωριστό κανάλι που τροφοδοτούταν σε ένα ή περισσότερα subwoofer. Τα συστήματα αναπαραγωγής σπιτιού, ωστόσο, μπορεί να μην έχουν ένα ξεχωριστό subwoofer, οπότε σύγχρονοι surround αποκωδικοποιητές σπιτιού και συστήματα περιλαμβάνουν συχνά ένα σύστημα διαχείρισης μπάσου που επιτρέπει το μπάσο σε οποιοδήποτε κανάλι (κύριο ή LFE) να τροφοδοτείται μόνο στα ηχεία που μπορούν να χειριστούν χαμηλής συχνότητας σήματα. Το εξέχον σημείο εδώ είναι ότι το κανάλι LFE δεν είναι το "κανάλι subwoofer": μπορεί να μην υπάρχει subwoofer και, αν υπάρχει, μπορεί να χειρίζεται πολύ περισσότερα από μόνο εφέ.

### 3.6 AMBISONICS – PanAmbio

Ambisonics είναι μια σειρά τεχνικών καταγραφής και αναπαραγωγής που χρησιμοποιεί τεχνολογία πολυκαναλικής αναμίξεως που μπορεί να χρησιμοποιηθεί είτε ζωντανά ή στο στούντιο και το οποίο αναδημιουργεί την ηχητικό πεδίο όπως υπήρχε στο χώρο, σε αντίθεση με τα παραδοσιακά συστήματα surround, τα οποία μπορούν να δημιουργήσει μόνο ψευδαίσθηση του ηχητικού πεδίου, εάν ο ακροατής βρίσκεται σε ένα πολύ στενό γλυκό σημείο μεταξύ ηχείων. Οποιοσδήποτε αριθμός ηχείων σε οποιαδήποτε φυσική ρύθμιση μπορεί να χρησιμοποιηθεί για να αναδημιουργήσει ένα ηχητικό πεδίο. Με 6 ή περισσότερα ηχεία τοποθετημένα γύρω από ένα ακροατή, ένα 3-διαστάσεων ("periphonic" ή πλήρης-σφαίρας) ηχητικό πεδίο μπορεί να παρουσιαστεί. Η Ambisonics εφευρέθηκε από τον Michael Gerzon.

Η Panor-Ambiophonic (PanAmbio) 4.0/4.1 συνδυάζει ένα στερεοφωνικό δίπολο και μια crosstalk ακύρωση μπροστά και μια δεύτερη σειρά πίσω από τον ακροατή (συνολικά τέσσερα ηχεία) για 360 ° 2D αναπαραγωγή surround. Ηχογραφήσεις τεσσάρων καναλιών, ειδικά αυτές που περιέχουν binaural cues, δημιουργούν ηχογραφήσεις speaker-binaural surround ήχου 5.1 καναλιών, συμπεριλαμβανομένων των DVDs ταινιών, είναι συμβατές με την ανάμιξη του περιεχομένου του C-καναλιού στο μπροστινό ζεύγος ηχείων. Το 6.1 μπορεί να παίζει με την ανάμιξη SC στο πίσω ζεύγος.

## 4 3D AUDIO

Τα 3D εφέ ήχου είναι μια ομάδα ηχητικών εφέ που χειρίζονται την εικόνα που παράγεται από στερεοφωνικά ηχεία, ηχεία surround, συστοιχίες ηχείων ή ακουστικά. Αυτό συνεπάγεται συχνά την εικονική τοποθέτηση των πηγών ήχου οπουδήποτε στον 3 διάστατο χώρο, συμπεριλαμβανομένων των πίσω, πάνω ή κάτω από τον ακροατή.

Υπάρχουν διάφοροι τύποι των 3D εφέ ήχου:

- Εκείνοι που διευρύνουν μόνο το στερεοφωνικό εικόνα τροποποιώντας την πληροφορία φάσης.
- Εκείνοι που μπορούν να τοποθετήσουν τους ήχους έξω από το στερεοφωνική βάση.
- Εκείνοι που περιλαμβάνουν μια πλήρη 3D προσομοίωση.

### 4.1 ΔΙΑΠΛΑΤΥΝΣΗ STEREO

Διαπλάτυνση της στερεοφωνικής εικόνας μπορεί να επιτευχθεί με χειρισμό της σχέσης του κεντρικού σήματος  $M$  και του πλευρικού σήματος  $S$ :  $M = \{(L + R) / 2\}$ ,  $S = \{(L - R) / 2\}$ . Ένα θετικό μέρος του πλευρικού σήματος  $S$  τώρα τροφοδοτείται μέσα στο αριστερό κανάλι και ένα τμήμα με ανεστραμμένη τη φάση του στο δεξί κανάλι. Μερικά boomboxes διαθέτουν μια τέτοια διαδικασία.

Ένας άλλος τρόπος να εξετάσουμε αυτό το ίδιο αποτέλεσμα, χωρίς προέκταση ενός κεντρικού και πλευρικού σήματος από τα αριστερά και δεξιά σήματα, είναι να προσθέσετε απλά το αριστερό σήμα, ελαφρώς εξασθενημένο και με ανεστραμμένη φάση, στο δεξί κανάλι και αντίστροφα. Λαμβάνοντας αυτό ένα βήμα παραπέρα, μια μικρή καθυστέρηση (20-100ms) μπορεί να προστεθεί στο ανεστραμμένο σήμα πριν από την ανάμιξη του με το αρχικό για έξοδο, προσθέτοντας μια μικρή αντίληψη στο αποτέλεσμα.

### 4.2 COMPLETE 3D POSITIONAL AUDIO

Η 3D προσομοίωση είναι η πιο προηγμένη ομάδα των 3D εφέ ήχου. Χρησιμοποιώντας λειτουργίες μεταφοράς κεφαλής και αντήχηση, οι αλλαγές του ήχου στην πορεία του από την πηγή (συμπεριλαμβανομένων των ανακλάσεων στους τοίχους και τα πατώματα) στο αυτί του ακροατή μπορεί να προσομοιωθεί. Αυτές οι επιδράσεις περιλαμβάνουν τον εντοπισμό των πηγών ήχου πίσω, πάνω και κάτω από τον ακροατή.

Μερικές 3D τεχνολογίες μετατρέπουν επίσης binaural ηχογραφήσεις σε στερεοφωνικές ηχογραφήσεις. *MorrowSoundTrue3D* μετατρέπει binaural, στερεοφωνικό, 5,1 και άλλες μορφές σε 8,1 μονή και πολλαπλή ζώνη 3D εμπειρία ήχου σε πραγματικό χρόνο.

3D εφέ ήχου Θέσης εμφανίστηκε στη δεκαετία του 1990 σε PC και κονσόλες παιχνιδιών. Ως μέσο, διαδραστικά παιχνίδια θα επωφεληθούν ίσως περισσότερο από κάθε άλλο. Ωστόσο, αν και κάποιες τεχνολογίες φαίνεται να λειτουργούν καλύτερα από άλλες, 3D ήχος στα παιχνίδια εξακολουθεί να αναπαράγεται με ακρίβεια συνήθως μόνο μέσω των ακουστικών.

Τεχνικές 3D ήχου έχουν επίσης ενσωματωθεί στη μουσική και στη μουσική βίντεο-παιχνιδιών. Το ερευνητικό έργο *Audioscape*, παρέχει στους μουσικούς ένα real-time 3D

οπτικοακουστικό περιβάλλον συγγραφής και απόδοσης περιεχομένου, κατάλληλο για χρήση σε ζωντανές εφαρμογές.

Ένα site με κινούμενα σχέδια και θεωρία του συστήματος με τη χρήση του HRTF για τη δημιουργία 3D ήχου: ISVR Εικονική Ακουστική.

Η αληθινή αναπαράσταση του επιπέδου ανύψωσης για την αναπαραγωγή 3D σε ηχείο κατέστη δυνατή από την Ambisonics και Wave field synthesis Principle, MorrowSound True3D και A & G 3D-EST.

### 4.3 ΠΡΟΤΕΡΑΙΟΤΗΤΑ ΗΧΟΥ

Εάν το πρόγραμμα περιήγησης αναπαραγωγής δεν έχει τους πόρους για να παίξει όλους τους ενεργές ήχους, συνιστάται ότι το πρόγραμμα ταξινομεί τους ενεργούς ήχους σε μια λίστα χρησιμοποιώντας τα ακόλουθα είδη με τη σειρά:

φθίνουσα προτεραιότητα

για ήχους με προτεραιότητα > 0,5, αύξηση (ώρα-startTime)

μείωση της έντασης στη θέση θεατή (ένταση × "εξασθένηση της έντασης")

όπου προτεραιότητα είναι το πεδίο προτεραιότητας του κόμβου Sound, που τώρα αντιπροσωπεύει τον τρέχοντα χρόνο, startTime είναι το startTime πεδίο του κόμβου της ηχητικής πηγής που καθορίζεται στο πεδίο πηγής, και "εξασθένηση της έντασης» αναφέρεται στον πολλαπλασιαστή έντασης που προέρχονται από το γραμμικό ramp εξασθένησης ντεσιμπέλ μεταξύ εσωτερικού και εξωτερικού ελλειψοειδούς.

Είναι σημαντικό το βασικό είδος 2 να χρησιμοποιείται για ήχους υψηλής προτεραιότητας (event και cue), ώστε νέα cues να ακούγονται ακόμα και όταν το πρόγραμμα αναπαραγωγής είναι "γεμάτο" από τρέχοντες ενεργούς ήχους υψηλής προτεραιότητας. Το 2<sup>ο</sup> είδος ταξινόμησης δεν πρέπει να χρησιμοποιείται για ήχους συνήθους προτεραιότητας, έτσι επιλογή μεταξύ τους θα βασίζεται στο 3<sup>ο</sup> είδος ταξινόμησης (ένταση στη θέση του παρατηρητή).

Το πρόγραμμα αναπαραγωγής θα παίξει όσους ήχους μπορεί, από την αρχή αυτής της ταξινομημένης λίστας, δεδομένων των διαθέσιμων πόρων και επιτρεπτής καθυστέρησης μεταξύ της απόδοσης. Στα περισσότερα συστήματα, οι πόροι που διατίθενται για MIDI streams είναι διαφορετικοί από εκείνους για το ήχους δειγμάτων, έτσι μπορεί να είναι κερδοφόρο να διατηρήσει μια ξεχωριστή λίστα για να χειριστεί δεδομένα MIDI.

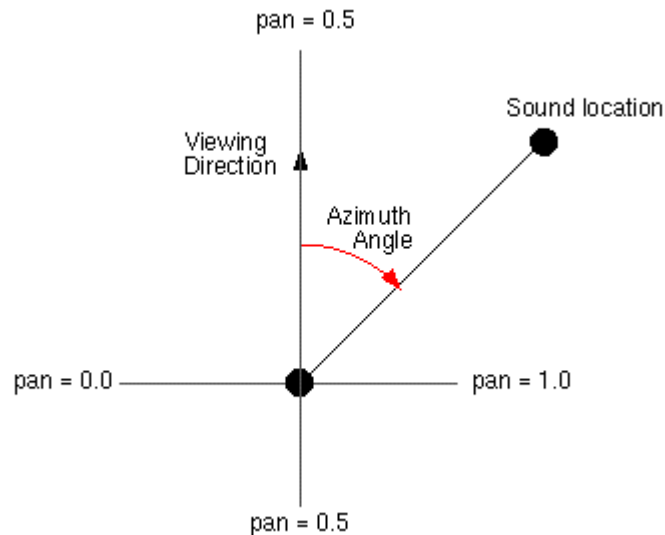
### 4.4 ΕΞΑΣΘΕΝΗΣΗ ΗΧΟΥ ΚΑΙ SPATIALIZATION

Προκειμένου να δημιουργηθεί μία γραμμική μείωση στην ένταση καθώς ο θεατής κινείται από το εσωτερικό προς το εξωτερικό ελλειψοειδές του ήχου, η εξασθένηση πρέπει να βασίζεται σε ένα γραμμικό ramp ντεσιμπέλ. Για να κάνετε η πτώση συνεπής σε όλα τα προγράμματα αναπαραγωγής, το ramp ντεσιμπέλ πρέπει να κυμαίνεται από 0 dB στο ελάχιστο ελλειψοειδές έως -20 dB στο εξωτερικό ελλειψοειδές. Οι κόμβοι sound με ένα εξωτερικό ελλειψοειδές, που είναι δέκα φορές μεγαλύτερο από το ελάχιστο θα εμφανιστεί το αντίστροφο τετράγωνο της έντασης πτώσης (drop-off) που προσεγγίζει εξασθένηση του ήχου σε περιβάλλον χωρίς ηχώ.

Τα προγράμματα αναπαραγωγής μπορούν να υποστηρίξουν χωρικό εντοπισμό των ήχων των οποίων ο χωρικός τομέας είναι TRUE, καθώς και οι βασικές βιβλιοθήκες τους θα το επιτρέψουν.

Προγράμματα αναπαραγωγής θα υποστηρίζουν τουλάχιστον στερεοφωνικό panning των μη-MIDI ήχων με βάση τη γωνία μεταξύ του θεατή και της πηγής. Η γωνία αυτή λαμβάνεται με την προβολή της θέσης ήχου (σε παγκόσμιο χώρο) επάνω στο επίπεδο XZ του θεατή. Προσδιορίζεται η γωνία μεταξύ του άξονα Z και του διανύσματος από το θεατή στη μετασχηματισμένη θέση, και εκχωρείται μια τιμή pan στην περιοχή [0.0, 1.0], όπως απεικονίζεται στην Εικόνα 1. Δεδομένης αυτής της pan αξίας, τα επίπεδα αριστερού και δεξιού καναλιού μπορούν να ληφθούν χρησιμοποιώντας τις ακόλουθες εξισώσεις:

$$\text{leftPanFactor} = 1 - \text{pan}^2$$
$$\text{rightPanFactor} = 1 - (1 - \text{pan})^2$$



**Εικόνα 1 : Stereo Panning**

Χρησιμοποιώντας την τεχνική αυτή, η ένταση του ήχου τροποποιείται από την τιμή πεδίου έντασης, και μετά από εξασθένηση λόγω απόστασης για να ληφθεί η unspatialized έξοδος ήχου. Οι τιμές στην unspatialized έξοδο ήχου στη συνέχεια κλιμακώνονται από το leftPanFactor και rightPanFactor για να προσδιοριστούν τα τελικά αριστερά και δεξιά σήματα εξόδου. Η χρήση των πιο εξελιγμένων τεχνικών εντοπισμού ενθαρρύνεται, αλλά δεν απαιτείται.

## 4.5 AUDIO CLIP

Ένας κόμβος AudioClip καθορίζει δεδομένα ήχου που μπορούν να αναφερθούν από κόμβους Sound.

Το πεδίο περιγραφής καθορίζει μια περιγραφή κειμένου της πηγής ήχου. Ένα πρόγραμμα αναπαραγωγής δεν απαιτείται να εμφανίσει το πεδίο περιγραφής, αλλά μπορεί να επιλεγεί να γίνει εκτός από την αναπαραγωγή του ήχου.

Το πεδίο url διευκρινίζει τη διεύθυνση URL από την οποία ο ήχος φορτώνεται. Τα προγράμματα αναπαραγωγής πρέπει να υποστηρίζουν τουλάχιστον τη wavefile μορφή σε ασυμπίεστη μορφή PCM. Συνιστάται, επίσης, τα προγράμματα αναπαραγωγής να υποστηρίζουν το αρχείο τύπου MIDI 1 μορφή ήχου και τη MP3 συμπίεσμένη μορφή. Τα αρχεία MIDI υποτίθεται ότι χρησιμοποιούν το General MIDI patch set. 9.2.1. Διευθύνσεις URL περιέχουν λεπτομέρειες σχετικά με το πεδίο url.



## 4.6 SOUND NODE

Ο κόμβος ήχου καθορίζει τη χωρική παρουσίαση του ήχου σε μία 3D σκηνή. Ο ήχος βρίσκεται σε ένα σημείο στο τοπικό σύστημα συντεταγμένων και εκπέμπει ήχο σε μια ελλειπτική τροχιά (που ορίζεται από δύο ελλειψοειδή). Τα ελλειψοειδή είναι προσανατολισμένα σε μια κατεύθυνση που καθορίζεται από το πεδίο κατεύθυνσης. Το σχήμα των ελλειψοειδών μπορεί να τροποποιηθεί για να παρέχει περισσότερο ή λιγότερο κατευθυντική εστίαση από τη θέση του ήχου.

Το πεδίο πηγής διευκρινίζει την πηγή ήχου για τον κόμβο ήχου. Αν το πεδίο πηγής δεν έχει καθοριστεί, ο κόμβος ήχου δεν θα εκπέμπει ήχο. Το πεδίο πηγής καθορίζει είτε ένα κόμβο AudioClip ή έναν κόμβο MovieTexture. Αν ένας κόμβος MovieTexture καθορίζεται ως πηγή ήχου, η MovieTexture αναφέρεται σε μια μορφή αρχείου που υποστηρίζει ήχο (ΠΑΡΑΔΕΙΓΜΑ MPEG-1Systems).

Το πεδίο έντασης ρυθμίζει την ένταση του ήχου (ντεσιμπέλ) που εκπέμπεται από τον κόμβο ήχου. Το πεδίο έντασης έχει μια τιμή που κυμαίνεται 0.0 έως 1.0 και καθορίζει ένα στοιχείο το οποίο θα πρέπει να χρησιμοποιείται για την κλιμάκωση των κανονικοποιημένων δεδομένων της ηχητικής πηγής κατά τη διάρκεια της αναπαραγωγής. Ένας κόμβος ήχου με ένταση 1.0 εκπέμπει ήχο με τη μέγιστη έντασή του (πριν από την εξασθένηση), και ένας κόμβος ήχου με ένταση 0.0 δεν εκπέμπει κανένα ήχο. Μεταξύ αυτών των τιμών, η ένταση θα πρέπει να αυξηθεί γραμμικά από μια αλλαγή -20 dB που πλησιάζει μια ένταση 0.0 σε μια μεταβολή 0 dB σε μία ένταση 1.0.

Το πεδίο προτεραιότητας παρέχει μια βοήθεια για το πρόγραμμα αναπαραγωγής για να επιλέξει τον ήχο τον οποίο θα παίξει, όταν υπάρχουν περισσότεροι ενεργοί κόμβοι ήχου από ό, τι μπορεί να παιχτεί ταυτόχρονα είτε λόγω περιορισμένων πόρων του συστήματος ή του φορτίου του συστήματος.

Το πεδίο τοποθεσίας καθορίζει τη θέση του πομπού του ήχου στο τοπικό σύστημα συντεταγμένων. Η έξοδος ενός κόμβου ήχου ακούγεται μόνο αν είναι μέρος της σκηνής.

Ο κόμβος ήχου έχει ένα εσωτερικό ελλειψοειδές που ορίζει έναν όγκο του χώρου στον οποίο η μέγιστη στάθμη του ήχου οριοθετείται. Μέσα σε αυτό το ελλειψοειδές, τα κανονικοποιημένα δεδομένα δείγματος κλιμακώνονται από το πεδίο έντασης και δεν υπάρχει καμία εξασθένηση. Το εσωτερικό ελλειψοειδές ορίζεται από την επέκταση του διανύσματος της κατεύθυνσης μέσω της θέσης. Τα πεδία minBack και minFront προσδιορίζουν αποστάσεις πίσω και μπροστά από τη θέση κατά μήκος του διανύσματος διεύθυνσης αντίστοιχα. Το εσωτερικό ελλειψοειδές έχει μία από τις εστίες του στη θέση (η δεύτερη εστία είναι σιωπηλή) και τέμνει το διάνυσμα κατεύθυνσης στο minBack και minFront.

Ο κόμβος ήχου έχει ένα εξωτερικό ελλειψοειδές που ορίζει ένα όγκο του χώρου που περιβάλλει την ακουστικότητα του ήχου. Δεν υπάρχει ήχος που μπορεί να ακουστεί έξω από αυτό το εξωτερικό ελλειψοειδές. Το εξωτερικό ελλειψοειδές ορίζεται από την επέκταση του διανύσματος κατεύθυνσης μέσω της θέσης. Τα πεδία maxBack και maxFront προσδιορίζουν αποστάσεις πίσω και μπροστά από τη θέση κατά μήκος του διανύσματος διεύθυνσης αντίστοιχα. Το εξωτερικό ελλειψοειδές έχει μία από τις εστίες του στη θέση (η δεύτερη εστία είναι σιωπηλή) και τέμνει το διάνυσμα κατεύθυνσης στο maxBack και maxFront.

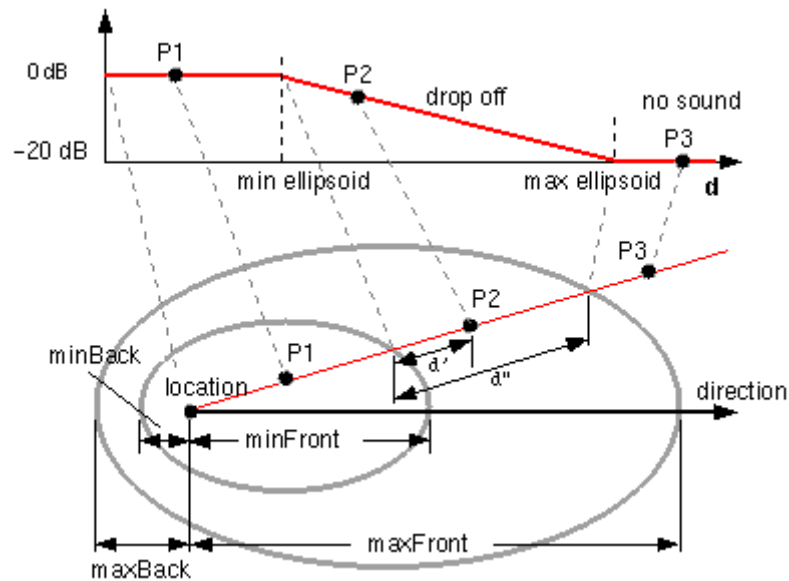
Τα minFront, maxFront, minBack και maxBack πεδία ορίζονται σε τοπικές συντεταγμένες και πρέπει να είναι μεγαλύτερες ή ίσες με το μηδέν. Το πεδίο minBack πρέπει να είναι μικρότερο ή ίσο με το maxBack, και το minFront πρέπει να είναι μικρότερο από ή ίσο με το maxFront. Οι παράμετροι του ελλειψοειδούς ορίζονται στο τοπικό σύστημα συντεταγμένων, αλλά η γεωμετρία των ελλειψοειδών επηρεάζεται από τους μετασχηματισμούς.

Μεταξύ των δύο ελλειψοειδών, θα υπάρχει μια γραμμική ramp εξασθένησης σε ένταση, από 0 dB στο ελάχιστο ελλειψοειδές έως -20 dB στο μέγιστο ελλειψοειδές:

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

όπου  $d'$  είναι η απόσταση κατά μήκος του διανύσματος θέσης-προς-θεατή, μετρούμενο από το μετασχηματισμένο ελάχιστο όριο ελλειψοειδούς προς το θεατή, και  $d''$  είναι η απόσταση κατά

μήκος του διανύσματος θέσης-προς-θεατή από το μετασχηματισμένο ελάχιστο όριο ελλειψοειδούς προς το μετασχηματισμένο μέγιστο όριο ελλειψοειδούς (βλέπε Εικόνα 2).



**Εικόνα 2 : Γεωμετρία Κόμβου Ήχου**

Το πεδίο *spatialize* καθορίζει αν ο ήχος γίνεται αντιληπτός ως κατευθυνσιακά τοποθετημένος σε σχέση με τον θεατή. Εάν το πεδίο *spatialize* είναι TRUE και ο θεατής βρίσκεται μεταξύ του μετασχηματισμένου εσωτερικού και του εξωτερικού ελλειψοειδούς, η κατεύθυνση του θεατή και η σχετική θέση του κόμβου ήχου θα πρέπει να ληφθούν υπόψη κατά την αναπαραγωγή. Αν το πεδίο *spatialize* είναι FALSE, τα αποτελέσματα κατεύθυνσης αγνοούνται, αλλά οι ελλειψοειδές διαστάσεις και η ένταση θα επηρεάσουν ακόμα την ένταση του ήχου. Αν η ηχητική πηγή είναι πολλαπλών καναλιών (για παράδειγμα στερεοφωνική), η πηγή διατηρεί διαχωρισμό των καναλιών της κατά τη διάρκεια της αναπαραγωγής.



## 5 OPEN AL

Το OpenAL (Open Audio Library) είναι ένα cross-platform API ήχου. Είναι σχεδιασμένο για την αποτελεσματική απόδοση του πολυκάναλου τρισδιάστατου στο χώρο ήχου. Το στυλ της διεπαφής και οι συμβάσεις του επίτηδες μοιάζουν με αυτά του OpenGL. Οι πρώτες εκδόσεις του πλαισίου εργασίας ήταν λογισμικό ανοιχτού κώδικα, αλλά οι μεταγενέστερες αναθεωρήσεις είναι ιδιόκτητες.

### 5.1 ΔΟΜΗ ΚΑΙ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ ΤΟΥ API

Η γενική λειτουργικότητα του OpenAL είναι κωδικοποιημένη σε αντικείμενα πηγής (source objects), ρυθμιστικά ήχου (audio buffers) και ένα μόνο ακροατή. Ένα αντικείμενο πηγής περιλαμβάνει ένα δείκτη προς ένα ρυθμιστικό (buffer), την ταχύτητα, τη θέση και την κατεύθυνση του ήχου και την ένταση του ήχου. Το αντικείμενο του ακροατή περιέχει την ταχύτητα, τη θέση και την κατεύθυνση του ακροατή και τη γενική ένταση όλου του ήχου. Οι buffers περιέχουν δεδομένα ήχου σε μορφή PCM, είτε 8 ή 16-bit, είτε σε μονοφωνική ή στερεοφωνική μορφή. Η μηχανή απόδοσης εκτελεί όλους τους απαραίτητους υπολογισμούς όσον αφορά την εξασθένηση λόγω απόσταση, φαινόμενο Doppler, κλπ.

Όταν αρχικοποιείται η OpenAL, τουλάχιστον μία συσκευή πρέπει να ανοίξει. Εντός της εν λόγω συσκευής, τουλάχιστον ένα πλαίσιο (context) θα δημιουργηθεί. Στο πλαίσιο αυτό, ένα αντικείμενο ακροατή υποδηλώνεται, και ένα πλήθος αντικειμένων πηγής (sources) μπορούν να δημιουργηθούν. Κάθε πηγή μπορεί να έχει ένα ή περισσότερα αντικείμενα ρυθμιστικά (buffers) που επισυνάπτονται σε αυτήν. Ρυθμιστικά αντικείμενα δεν αποτελούν μέρος του ενός συγκεκριμένου πλαισίου - μοιράζονται ανάμεσα σε όλα τα πλαίσια σε μία συσκευή.

Το καθαρό αποτέλεσμα όλων αυτών για τον τελικό χρήστη είναι ότι σε μία κατάλληλα γραμμένη OpenAL εφαρμογή, οι ήχοι συμπεριφέρονται αρκετά φυσικά καθώς ο χρήστης κινείται διαμέσου του τρισδιάστατου χώρου του εικονικού κόσμου. Από την πλευρά του προγραμματιστή, πολύ λίγη πρόσθετη εργασία απαιτείται για να συμβεί αυτό σε μια υπάρχουσα βάση OpenGL 3D γραφική εφαρμογή.

Σε αντίθεση με την προδιαγραφή OpenGL, η OpenAL προδιαγραφή περιλαμβάνει δύο τμήματα του API: ο πυρήνας που αποτελείται από τις πραγματικές κλήσεις λειτουργία OpenAL, και το ALC (Audio Library Context) API το οποίο χρησιμοποιείται για τη διαχείριση πλαισίων απόδοσης, τη χρήση των πόρων και κλείδωμα κατά έναν ανεξάρτητης πλατφόρμας τρόπο. Υπάρχει, επίσης, μία «ALUT» βιβλιοθήκη που παρέχει ανώτερες «βολικές» λειτουργίες - ακριβώς ανάλογη με τη «GLUT» του OpenGL.

Με σκοπό να παρέχει πρόσθετη λειτουργικότητα στο μέλλον, η OpenAL χρησιμοποιεί έναν μηχανισμό επέκτασης. Μεμονωμένοι πωλητές είναι έτσι σε θέση να συμπεριλάβουν τις δικές τους επεκτάσεις σε διανομές OpenAL, συνήθως για το σκοπό της έκθεσης πρόσθετης λειτουργικότητας σε ιδιόκτητο εξοπλισμό τους. Οι επεκτάσεις μπορούν να προωθηθούν με ARB (Architecture Review Board) κατάσταση, υποδεικνύοντας μία πρότυπη επέκταση η οποία θα διατηρηθεί για προς τα πίσω συμβατότητα. Οι επεκτάσεις ARB έχουν την προοπτική να προστεθούν στο API του πυρήνα μετά από ένα χρονικό διάστημα.

Για προηγμένη ψηφιακή επεξεργασία σήματος και hardware-accelerated ηχητικά εφέ, μπορούν να χρησιμοποιηθούν το EFX (Effects Extension) ή environmental audio extensions (EAX).

## 5.2 ΜΕΙΟΝΕΚΤΗΜΑΤΑ

Το ενιαίο μοντέλο ακροατή σε OpenAL είναι προσαρμοσμένη σε ένα μόνο χρήστη και δεν είναι κατάλληλο για την τεχνητή νοημοσύνη ή ρομπότ ή πολλούς συμμετέχοντες σε μουσικές παραστάσεις. Σε αυτές τις περιπτώσεις ένα μοντέλο πολλαπλών ακροατών απαιτείται. Η OpenAL αποτυγχάνει επίσης να ληφθούν υπόψη οι καθυστερήσεις διάδοσης ήχου (η ταχύτητα του ήχου χρησιμοποιείται για την επίδραση του εφέ Doppler μόνο). Η απόσταση από μία πηγή ήχου μεταφράζεται μόνο σε ένα αποτέλεσμα πλάτους (εξασθένησης) και όχι μια καθυστέρηση. Ως εκ τούτου η OpenAL δεν μπορεί να χρησιμοποιηθεί για τη διαφορά χρόνου των υπολογισμών άφιξης, εκτός εάν μπορείτε να γράψετε μόνοι σας αυτήν τη λειτουργικότητα.

## 5.3 ΦΟΡΗΤΟΤΗΤΑ

Το API είναι διαθέσιμο στις ακόλουθες πλατφόρμες:

- Λειτουργικά συστήματα :
  - AmigaOS 3.x
  - BSD
  - Linux, για OSS και ALSA backends
  - iOS
  - IRIX
  - Mac OS 8/9
  - Mac OS X
  - Microsoft Windows
  - MorphOS
  - OpenBSD
  - BlackBerry PlayBook
  - Solaris
- Κονσόλες βίντεο παιχνιδιών:
  - Xbox
  - Xbox 360
- Φορητές κονσόλες παιχνιδιών:
  - PlayStation Portable

## 6 Η ΕΦΑΡΜΟΓΗ ΜΑΣ (AUDIO PLAYER)

Η εφαρμογή μας αποτελείται από τα εξής δομικά στοιχεία:

- Open, play, stop buttons
- 5 jCheckBoxes, 4 για τον προσδιορισμό ηχείου αναπαραγωγής και 1 για ενεργοποίηση/απενεργοποίηση looping (επανάληψης του ήχου)
- 5 jSliders, 3 για τον προσδιορισμό της θέσης αναπαραγωγής του ήχου στους 3 άξονες x,y,z, 1 slider για αυξομείωση της έντασης και 1 για την αυξομείωση του τόνου (pitch).
- 1 jTextArea όπου αναφέρονται οι κινήσεις του χρήστη



Εικόνα 3 : Audio Player

### 6.1 JFILECHOOSER

Πατώντας το openButton ανοίγουμε ένα καινούριο παράθυρο του filechooser ο οποίος έχει οριστεί στον constructor της κλάσης μας.

```

public Audioplayer() { //Create a file chooser
    fc = new JFileChooser();
    fc.addChoosableFileFilter(new SoundFilter());
    fc.setAcceptAllFileFilterUsed(false);
    initComponents();
}

```

### Ορισμός του filechooser

Εδώ, εκτός απ' τη δημιουργία του filechooser μας, βλέπουμε πως ορίζουμε τι τύπου αρχεία θα ανοίγει, χρησιμοποιώντας την addChoosableFileFilter, η οποία καλεί τη SoundFilter, στην οποία γίνεται η επιλογή του τύπου αρχείου. Επίσης απενεργοποιούμε τη δυνατότητα να μας δείχνει όλων των τύπων τα αρχεία το openFileDialog μας κάνοντας setAcceptAllFileFilterUsed(false);

```

private void
openButtonActionPerformed(java.awt.event.ActionEvent evt)
{
    int returnVal = fc.showOpenDialog(Audioplayer.this);
    ...
}

```

### Άνοιγμα καινούριου παραθύρου filechooser

```

public class SoundFilter extends FileFilter {
public boolean accept(File f) {
    if (f.isDirectory()) {return true;}
    String extension = Utils.getExtension(f);
    if (extension != null) {
        if (extension.equals(Utils.wav)) {
            return true;
        } else {
            return false;
        }
    }
    return false;
}
public String getDescription() {
    return ".wav";
}
}

```

### SoundFilter.java

Μ' αυτόν τον τρόπο παίρνουμε ως όρισμα το αρχείο που θέλουμε να φορτώσουμε για αναπαραγωγή και το τυπώνουμε στο JTextArea μας.

```
private void openButtonActionPerformed(java.awt.event.ActionEvent evt) {  
    ...  
    if (returnVal == JFileChooser.APPROVE_OPTION) {  
        File file = fc.getSelectedFile();  
        JTextArea1.append(file.getName() + " loaded." + newline);  
        path = file.getPath();  
  
        System.out.println(path.toString());  
  
    } else {  
        JTextArea1.append("Open command cancelled by user." + newline);  
    }  
    JTextArea1.setCaretPosition(jTextArea1.getDocument().getLength());  
}
```

#### Αρχικοποίηση του αρχείου



Εικόνα 4 : Τυπώνουμε το αρχείο που φορτώσαμε

## 6.2 PLAY, STOP BUTTON

Πατώντας το playButton τρέχουμε ένα thread, το οποίο έχουμε ορίσει στα αντικείμενα της κλάσης μας και είναι τύπου PlayThread.

```
public class Audioplayer extends  
    javax.swing.JFrame {  
    ...PlayThread thread = new  
    PlayThread();... }
```

### Ορισμός thread

```
private void  
playButtonActionPerformed(java.awt.event.ActionEvent evt) {  
    thread.start();  
}
```

### Ξεκίνημα Thread

Στο thread μας ανοίγουμε μια συσκευή OpenAL και ορίζουμε μια πηγή που παίρνει ως όρισμα το path του αρχείου που θέλουμε να αναπαράγουμε. Το path έχει αρχικοποιηθεί κι όλας από το openButton. Με τον ορισμό του thread αντικειμένου μας μπορούμε να καλούμε τις μεθόδους της ηχητικής πηγής (thread.source.stop, thread.source.setPosition κτλ). Έτσι έχουμε φορτώσει το αρχείο μας και είναι έτοιμο για αναπαραγωγή.

Τώρα μπορούμε να αναπαράγουμε το αρχείο, να ελέγξουμε τη θέση της ηχητικής πηγής (setPosition(float x,float y,float z)), την ένταση (setGain()), το pitch (setPitch()).



```

public class PlayThread extends Thread{
    Source source;
    private Component frame;
    private int i;

    //private Buffer j;
    public PlayThread() {
super();
    }
    public void run() {
        try {
            if (Audioplayer.path==null){
                JOptionPane.showMessageDialog(frame, "Select a file first.");
            }

            OpenAL openal = new OpenAL();
            source = openal.createSource(new File(Audioplayer.path));
            System.out.println(source.toString());
            source.play();
            source.setGain(0.75f);    // 75% volume
            source.setPitch(0.85f);  // 85% of the original pitch
            source.setPosition(0, 0, 0); // -1 means 1 unit to the left
            source.setLooping(false); // Loop the sound effect
            //j=source.getBuffer();
            //System.out.println(j);
            for (i=1;i<=10000;i++){
                Thread.sleep(1);    // Wait for 10 seconds
            }
            //Thread.sleep(10000);    // Wait for 10 seconds
            source.close();
            openal.close();
        } catch (InterruptedException ex) {
            Logger.getLogger(PlayThread.class.getName()).log(Level.SEVERE, null,
ex);
        } catch (IOException ex) {
            Logger.getLogger(PlayThread.class.getName()).log(Level.SEVERE, null,
ex);
        } catch (UnsupportedAudioFileException ex) {
            Logger.getLogger(PlayThread.class.getName()).log(Level.SEVERE, null,
ex);
        } catch (ALException ex) {
            Logger.getLogger(PlayThread.class.getName()).log(Level.SEVERE, null,
ex);
        }
    }
}
}

```

PlayThread.java

Πατώντας το stopButton κλείνουμε την πηγή και την OpenAL συσκευή που είχαμε ανοίξει με το playButton.

```
private void
stopButtonActionPerformed(java.awt.event.ActionEvent evt)
{
    thread.source.close();
    openal.close();
}
```

### Stop Button

## 6.3 X, Y, Z POSITION

Κατά τη διάρκεια της αναπαραγωγής ενεργοποιείται η δυνατότητα επιλογής του ηχείου αναπαραγωγής και ενεργοποίησης/απενεργοποίησης του looping (επανάληψης του ήχου).

Χρησιμοποιούμε γι' αυτό τη συνάρτηση setPosition(float x,float y,float z),στο ItemEvent των checkboxes left, right, front, rear, όπου ο κάθε float αριθμός μας δείχνει πόσες μονάδες δεξιά ή αριστερά (x άξονας), μπρος ή πίσω (y άξονας), αναλόγως το πρόσημο, θα είναι μετακινημένη η θέση της ηχητικής πηγής. Έτσι έχουμε :

```
private void jCheckBox1ItemStateChanged(java.awt.event.ItemEvent evt) {
    Object sourceBox = evt.getItemSelectable();
    thread.source.setPosition(-2f, 0, 0);
    if (evt.getStateChange() == ItemEvent.DESELECTED) {
        thread.source.setPosition(0, 0, 0);
    }
}
```

### setPosition to the Left

όπου με την εντολή thread.source.setPosition(-2f, 0, 0); μετακινούμε την ηχητική πηγή 2 μονάδες προς τα αριστερά στον άξονα του x.

```
private void jCheckBox3ItemStateChanged(java.awt.event.ItemEvent evt) {
    Object sourceBox = evt.getItemSelectable();
    thread.source.setPosition(2f, 0, 0);
    if (evt.getStateChange() == ItemEvent.DESELECTED) {
        thread.source.setPosition(0, 0, 0);
    }
}
```

### setPosition to the Right

Αντίστοιχα με την εντολή `thread.source.setPosition(2f, 0, 0)`; μετακινούμε την ηχητική πηγή 2 μονάδες προς τα δεξιά στον άξονα του x.

Με τον ίδιο τρόπο μετακινούμε την πηγή στους άξονες y και z

```
private void jCheckBox4ItemStateChanged(java.awt.event.ItemEvent evt) {
    Object sourceBox = evt.getItemSelectable();
    thread.source.setPosition(0, 2f, 0);
    if (evt.getStateChange() == ItemEvent.DESELECTED) {
        thread.source.setPosition(0, 0, 0);
    }
}
```

#### **setPosition to the Front**

```
private void
jCheckBox5ItemStateChanged(java.awt.event.ItemEvent evt) {
    Object sourceBox = evt.getItemSelectable();
    thread.source.setPosition(0, -2f, 0);
    if (evt.getStateChange() == ItemEvent.DESELECTED) {
        thread.source.setPosition(0, 0, 0);
    }
}
```

#### **setPosition to the Rear**

Επίσης για να δώσουμε τιμή στους float αριθμούς της μεθόδου `setPosition` μπορούμε να χρησιμοποιήσουμε `jSliders`, ένα για τον κάθε άξονα.

```
private void xPositonStateChanged(javax.swing.event.ChangeEvent evt) {
    JSlider sourceSlider = (JSlider)evt.getSource();
    if (sourceSlider.getValueIsAdjusting()) {
        float value = (float)sourceSlider.getValue();
        System.out.println(value);
        thread.source.setPosition(value/100f,0,0);
    }
}
```

#### **jSlider setPosition x**

```

private void
yPositionStateChanged(javax.swing.event.ChangeEvent
evt) {
    JSlider sourceSlider = (JSlider)evt.getSource();
    if (sourceSlider.getValueIsAdjusting()) {
        float value = (float)sourceSlider.getValue();
        System.out.println(value);
        thread.source.setPosition(0,value/100f,0);
    }
}

```

**jSlider setPosition y**

```

private void zPositionStateChanged(javax.swing.event.ChangeEvent evt) {
    JSlider sourceSlider = (JSlider)evt.getSource();
    if (sourceSlider.getValueIsAdjusting()) {
        float value = (float)sourceSlider.getValue();
        System.out.println(value);
        thread.source.setPosition(0,0,value/100f);
    }
}

```

**jSlider setPosition z**

## 6.4 GAIN – PITCH – LOOPING

Τέλος μπορούμε να ρυθμίσουμε την ένταση και τον τόνο (pitch) του ήχου χρησιμοποιώντας jSliders. Κάνουμε χρήση των μεθόδων setGain και setPitch αντίστοιχα.

```

private void gainSliderStateChanged(javax.swing.event.ChangeEvent
evt) {
    JSlider sourceSlider = (JSlider)evt.getSource();
    if (sourceSlider.getValueIsAdjusting()) {
        float value = (float)sourceSlider.getValue();
        System.out.println(value);
        thread.source.setGain(value/100f);
    }
}

```

**gainSlider**

```

private void
pitchSliderStateChanged(javax.swing.event.ChangeEvent evt) {
    JSlider sourceSlider = (JSlider)evt.getSource();
    if (sourceSlider.getValueIsAdjusting()) {
        float value = (float)sourceSlider.getValue();
        System.out.println(value);
        thread.source.setPitch(value/100f);
    }
}

```

### **pitchSlider**

Ακόμη, υπάρχει η χρήση ενός checkbox για την ενεργοποίηση ή όχι του looping (επανάληψη του ήχου). Η μέθοδος που χρησιμοποιούμε εδώ είναι η setLooping(true ή false).

```

private void jCheckBox6ItemStateChanged(java.awt.event.ItemEvent evt) {
    Object sourceBox = evt.getItemSelectable();
    thread.source.setLooping(true);
    if (evt.getStateChange() == ItemEvent.DESELECTED) {
        thread.source.setLooping(false);
    }
}

```

### **setLooping**

## **6.5 ΑΠΑΙΤΟΥΜΕΝΕΣ ΒΙΒΛΙΟΘΗΚΕΣ**

Για να τρέξει σωστά το πρόγραμμά μας χρειάζονται κάποιες βιβλιοθήκες τύπου .jar να προστεθούν στο project μας, για να χρησιμοποιήσουμε κάποιες απ' τις μεθόδους τους.

Αυτές είναι οι : jna.jar και η net.java.games.joal\_1.1.0.jar.

Ακόμη, χρειάζεται να αντιγράψουμε κάποια αρχεία .dll στο φάκελο C:\Program Files (x86)\Java\jdk1.6.0\_31\jre\lib\ext (τα OpenAL32.dll, OpenAL64.dll, soft\_oal.dll)

Επίσης στο ALFactory.java χρειάζεται να προσδιορίσουμε το path της soft\_oal.dll βιβλιοθήκης (private static final String DEFAULT\_DLL\_NAME = "C://Program Files (x86)/Java/jdk1.6.0\_31/jre/lib/ext/soft\_oal.dll";)

Τέλος σε περίπτωση που δε βρίσκει το πρόγραμμα το CopyLibs.jar, πάμε στο classpath από το Tools>Options>Miscellaneous>Ant και το προσθέτουμε (add JAR).

## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

- [1] [en.wikipedia.org](http://en.wikipedia.org)
- [2] [OpenAL Programmer's Guide.pdf](#)
- [3] [docs.oracle.com/oracle\\_documentation](http://docs.oracle.com/oracle_documentation)
- [4] [web3d.org](http://web3d.org) (sound specifications)