# Zirkonium MK III User Guide (ver 3.2)

ZKM | Institute for Music and Acoustics

Chikashi Miyama

ZKM | INSTITUTE FOR MUSIC AND ACOUSTICS, KARLSRUHE GERMANY

HTTP://ZKM.DE/INSTITUTE-FORSCHUNGSSTELLEN/INSTITUT-FUER-MUSIK-UND-AKUSTIK

*December 2016*

# Contents

# 1. Introduction

## 1.1 Overview

Zirkonium is a set of MacOSX software tools to aid the composition of spatial music; the software enables composers to design multiple spatial trajectories with an intuitive GUI and arrange them in time. According to the provided trajectory information, the actual audio signals can then be rendered in real-time for virtually any type of 2D or 3D loudspeaker system.

The Software was originally developed for the ZKM Klangdom (Sound Dome), a 3D surround audio system, consisting of 47 loudspeakers [Figure:1.1], but it is also utilized for any 2D and 3D loudspeaker systems.

For developing the latest version of Zirkonium, we focused on improving the aspects of usability, visualisation, efficiency, and compatibility. Consequently, the software structure and the GUI were entirely reassessed and redesigned. Furthermore, a number of functionalities, such as parametric trajectory generator, automatic interpolation, event filter, and SpatDIF-export, are newly implemented.

Figure 1.1: Klangdom installed in Kubus ZKM

## 1.2 Features

Zirkonium MK III has following features:

### Graphical manipulation of sound trajectories with bézier curves

Zirkonium facilitates the manipulation of highly complex sound trajectories. By multi-segmented-bézier-curves, you are able to draw sound paths intuitively and flexibly.

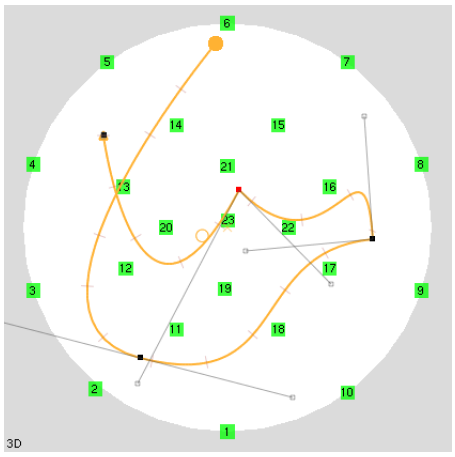


Figure 1.2: A sound trajectory drawn with bézier curves.

### Parameter-based Trajectory Creation

The software offers another approach for trajectory creation. With “Add circle / spiral” popover panel, you can generate spiral or circle-shaped trajectories algorithmically by inputting a few parameters.

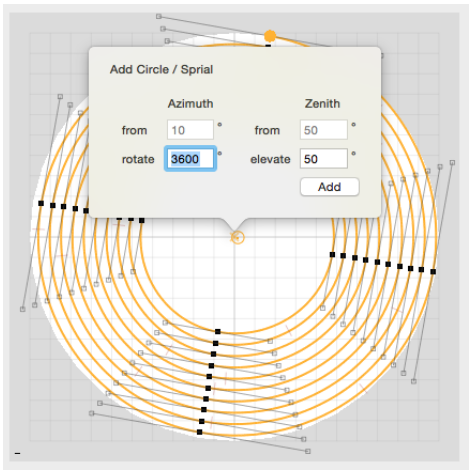


Figure 1.3: An example of parameter-based trajectory creation

### DAW-like Event Handling

All imported sound files are instantly analyzed and the waveforms of the files are displayed in the manner of DAW software. Furthermore, the software allows you to create special events graphically on top of these waveforms, so that you can grasp the relationship between audio contents and spatial events at a glance.

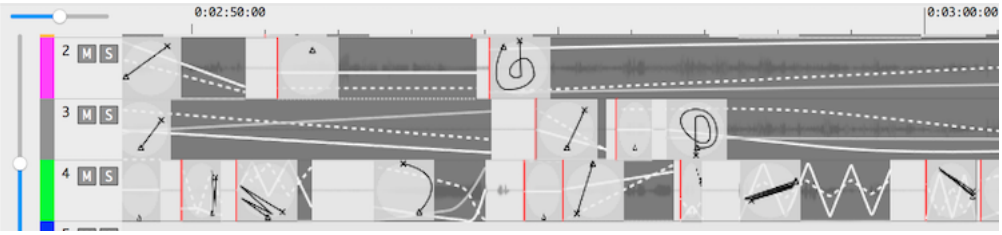


Figure 1.4: DAW-like Editing of Spatial Events

### Event Filtering

You may sometimes want to process a large number of spatial events at once. The powerful event filtering feature of Zirkonium enables you to instantly select events that match provided conditions and allows you to manipulate these events at once.


Figure 1.5: Editing the condition for event filtering

### Quick Event Manipulation

In Zirkonium, all events are editable with mouse operation, but it is also possible to edit them using text input with the **quick event manipulator**. This assists you in setting up the start time and end time of the events, or scaling the duration of the event with accuracy.

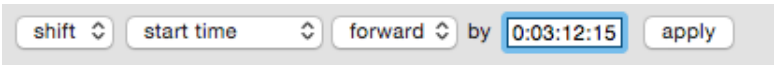


Figure 1.6: Inputting time for event time shifting in quick event manipulator

### Powerful Visualization of Spatialized Sound

All waveform images of audio contents are utilized also in Dome view and Motion view. In Dome view, the waveforms are rendered along sound paths, thus enabling you to grasp the relationship between audio contents and their positions in space.

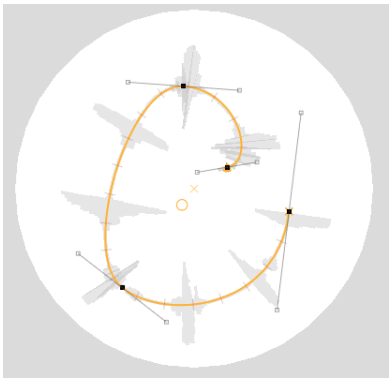

Figure 1.7: Waveform rendered along a sound path

### Variety of Rendering Algorithms

Zirkonium currently features two types of spatial rendering algorithms, VBAP and Ambisonics. In Zirkonium, you are able to assign an algorithm for each ID (sound object) and employ both algorithms simultaneously. For Ambisonics, further options for optimization (*in phase* and *maxRe* optimization) are available.

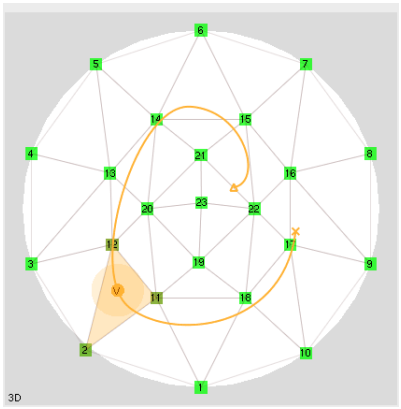


Figure 1.8: Domeview displays active VBAP speaker triplet during playback

### HRTF Simulation

With the HRTF (Head Related Transfer Function) algorithm, Zirkonium realizes a virtual 3D sound for headphone listening. This functionality enables you to continue working on your composition in any kind of environment.

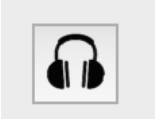

Figure 1.9: HRTF mode button

### Customization of Core Spatial Rendering Algorithm

In principle, the spatial rendering algorithms in Zirkonium is optimized for 3D surround systems, such as Klangdom. However, the software grants you an access to the core spatial algorithms, written in Pd (Pure Data). By modifying this core spatial algorithm, you can tune it for other surround systems.

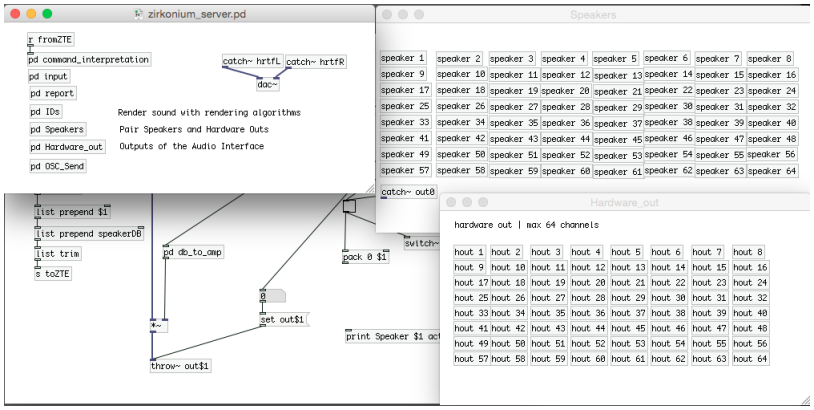


Figure 1.10: Spatial Rendering Algorithm is implemented as a Pd Patch

### SpatDIF Compatibility

Zirkonium has a functionality to export all spatial events to a SpatDIF conformed XML file. With this functionality, you can reuse trajectories and events created with Zirkonium in other software, such as Max or Pure Data.

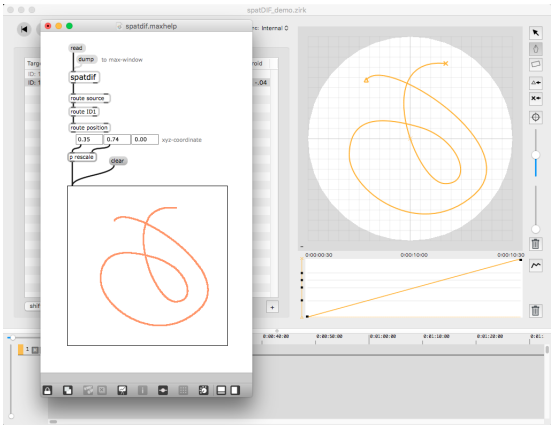


Figure 1.11: A trajectory drawn in Zirkonium imported by a Max patch

### Synchronization with other software through OSC or MTC

«««< HEAD Zirkonium is able to synchronize with other software that are compatible with OSC (Open Sound Control) or MTC (MIDI Time Code). This feature is useful, when we synchronize a video file running on another application, while playing back audio from Zirkonium. ======= Zirkonium is able to synchronize with other software that are compatible with OSC (Open Sound Control) or MTC (MIDI Time Code) as a master or a slave. This feature is useful, for example, to synchronize a video file running on another application, while playing back audio from Zirkonium. »»»> 044d455ba68f450b238e5d2a7d78f68046e2d50c

## 1.3 Architecture

Figure 1.12 illustrates the software architecture of Zirkonium Trajectory Editor. All GUIs are developed with Cocoa, OpenGL and GLSL. Apple's Core Data is utilized for data management.

The spatial rendering functionalities are programmed as a Pure Data patch. This Pd patch, as well as Pure Data sound processing engine, are embedded in the Zirkonium Trajectory Editor using libPd.

Audio signals, processed by embedded Pd, are sent to the audio hardware through Port Audio and Core Audio.

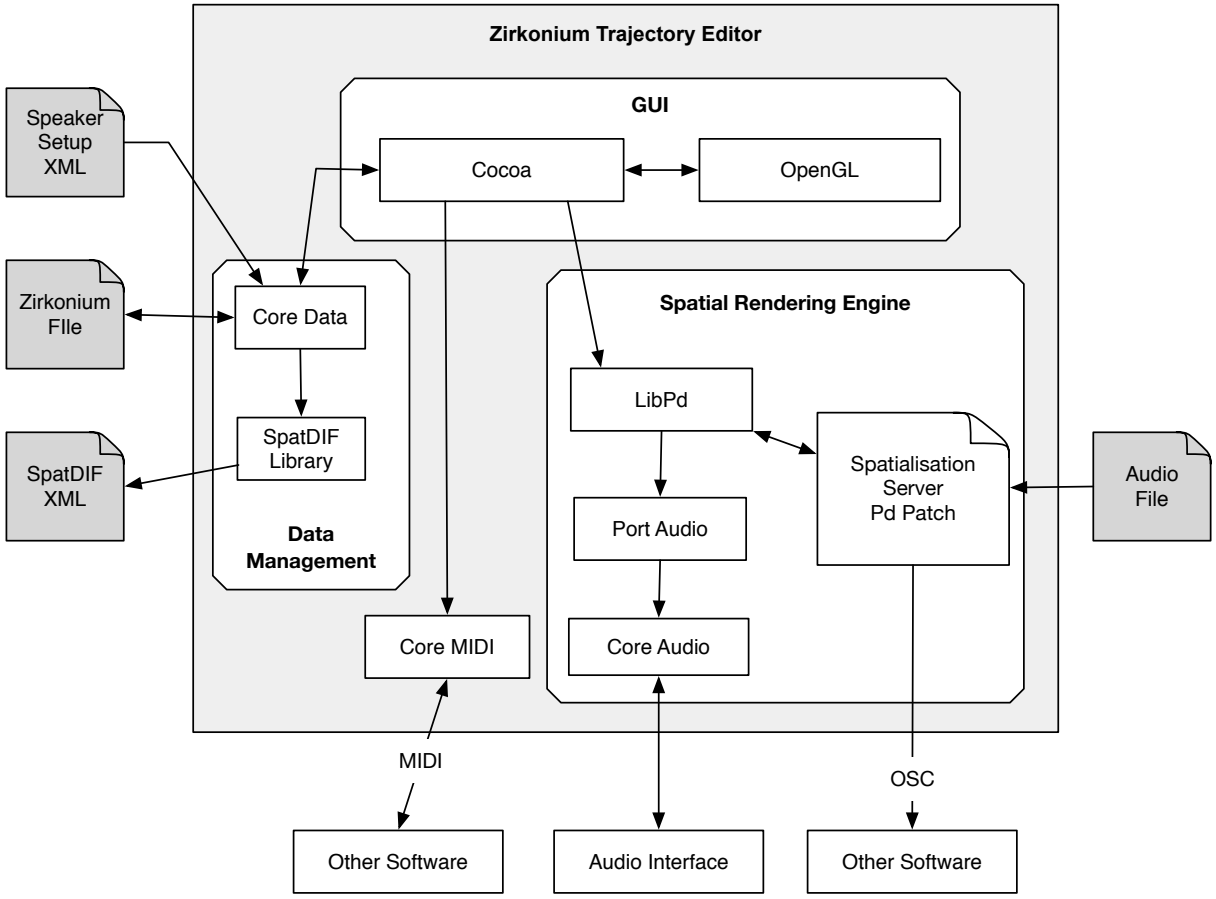


Figure 1.12: Software Architecture

## 1.4 Dependencies

Zirkonium employs following Open Source Libraries in addition to MacOSX Cocoa Framework:

## 1.5 System Requirements

The system requirement for Zirkonium is:

1. Operating System: 10.9 or higher
2. CPU : Intel Core i5 or i7
3. RAM : min. 4 GB

The software is tested and evaluated on Mac OS 10.11 (El Capitan).

| Name | License | Author | URL |
|---|---|---|---|
| Pure Data | Standard Improved BSD | Miller Puckette | `http://puredata.info` |
| LibPd | Standard Improved BSD | Peter Brinkmann *et al.* | `http://libpd.cc` |
| SpatDIF Library | MIT | Chikashi Miyama *et al.* | `http://spatdif.org` |
| Port Audio | MIT | Ross Bencina | `http://www.portaudio.com` |
| HOA Library | GPLv3 | Julien Colafrancesco *et al.* | `http://www.mshparisnord.fr` |

Table 1.1: Dependencies

## 1.6 What's in the package

In the Zirkonium package, you will find three separate applications.

1. Zirkonium Trajectory Editor
2. Speaker Setup
3. ZirkQTPlayer

Other than **Zirkonium Trajectory Editor**, the main application, Zirkonium package includes two more applications, **Speaker Setup** and **ZirkoniumQTPlayer**.

Speaker Setup is an application for configuring speaker setup in a 2D or 3D space. This application is able to export a XML file that defines speaker positions in a space. These XML files can be loaded onto the Dome View of the Zirkonium Trajectory Editor for defining your own speaker setup. For details see chapter 11

ZirkQTPlayer is a simple Quicktime player that is capable of receiving OSC Messages from Zirkonium Trajectory Editor. This software enables you to synchronize sound tracks that spatialized with Zirkonium and a Quicktime Movie. For details see chapter 12.

## 1.7 Installation

For the installation, simply copy all three applications to your Applications folder.

## 1.8 The Coordinate System of Zirkonium

There are many different coordinate systems for describing a position in a 3D space. Zirkonium Trajectory Editor and Speaker Setup application adopt the head-related coordinate system, defined by Jens Blauert in his book entitled Spatial Hearing: The Psychophysics of Human Sound Localization.

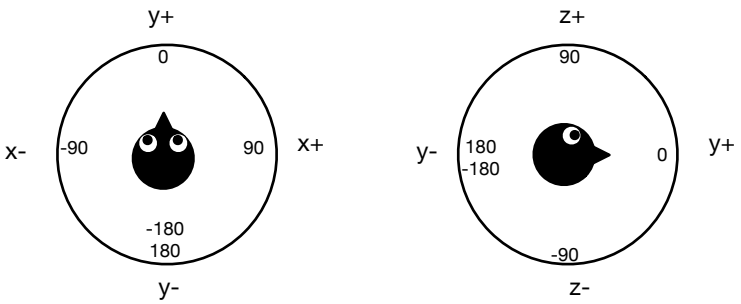


Figure 1.13: The Coordinate System of Zirkonium

In this coordinate system, the relationship between the listener and **Azimuth** is defined as follows:

- Front ... 0°
- Left ... -90°
- Back ... 180° or -180°
- Right ... 90°

The relationship between the listener and the **Elevation** is defined as follows:

- Above (Zenith) ... 90°
- Below (Nadir)... -90°

The relationship between the listener and the Cartesian axes is defined as follows:

- Left ... x-
- Right ... x+
- Front ... y+
- Back ... y-
- Above ... z+
- Below ... z-

Figure 1.13 depicts the relationship between the listener and both Spherical and Cartesian coordinates used in Zirkonium.

# 2. GUI overview

You can create, open or reopen files using the **File -> New / Open / Open Recent** options. After you create a new document, a new window should appear on screen [fig:2.1].

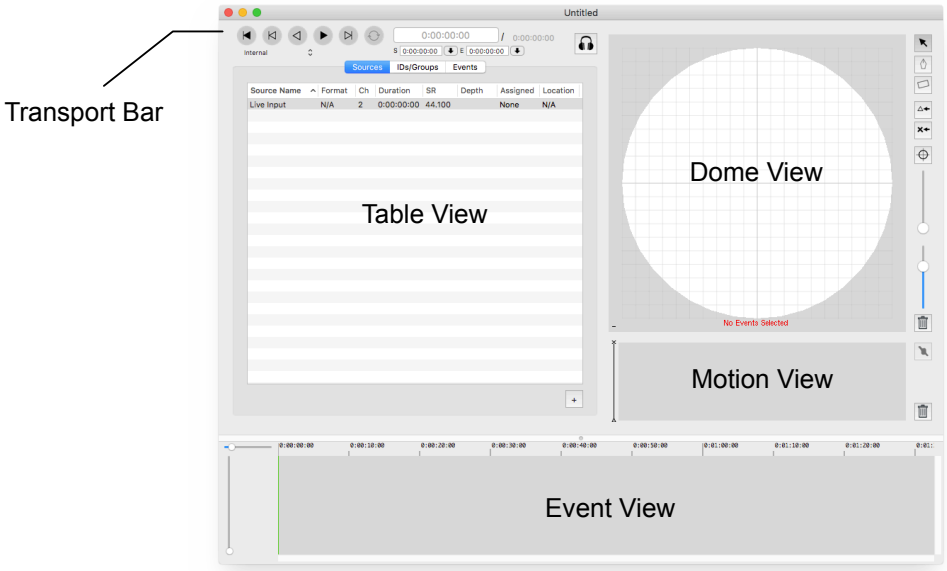


Figure 2.1: New window

The window consists of following five components.

- Table View
- Dome View
- Motion View

- Event View
- Transport Bar

In order to give a glimpse, this chapter describes the functionalities of each component briefly. For details, see chapters 4 - 7.

## 2.1 Table View

This view displays all data in the current project in tables. The view is comprised of three tabs; **Sources**, **IDs/Groups** and **Events**.


Figure 2.2: Table view tab

### Sources Tab

This is the start point of your spatial composition. The table in this tab lists all imported sound files. You can add new files to this list for playback or remove previously added files from the list.

### IDs/Groups Tab

This tab has two tables. The top table is called **IDs Table** and the bottom table is called **Groups Table**. Here you can define IDs (sound objects) and Groups of IDs. For details of ID and Group, refer to chapter 5 and 6.

### Events Tab

All events of IDs, Groups and Markers in the piece is listed here. You can add, delete or modify single or multiple events in this tab. At the bottom of the tab, there is a group of pop-up menus and text fields. This is called **Quick Event Manipulator**. This small GUI component enables you to quickly modify the properties of selected events.

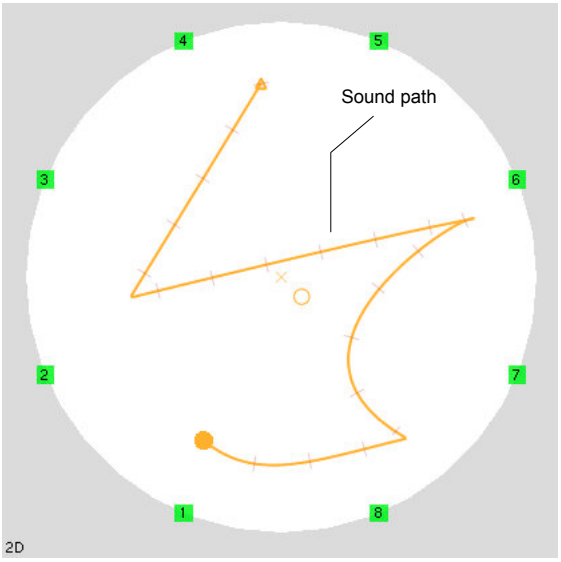


Figure 2.3: Sound path in the Dome view

## 2.2 Dome View

Dome View is a canvas for the **Sound Path**. Using bézier curves, you can design the paths of sound trajectory for each spatial event. During the playback, this view visualizes active sound paths, the level of audio signals sent to each loudspeaker, the movement of each ID, selected loudspeaker triplets by the VBAP algorithm etc. in real-time [fig:2.3].

## 2.3 Motion View

In the Dome View, you draw paths that sound objects (i.e. IDs) move along. Obviously, a sound path has a start and an end point. However, it does not necessarily mean that all IDs move from the start point to the end point at a *constant* speed. In Motion view, you can draw a **Motion path** that determines at what speed IDs move along the corresponding sound path. Furthermore, this view provides also the control over **Span Path** that represents the spatial size of each ID in time.


Figure 2.4: Motion path in the Motion view

## 2.4 Event View

This View shows all spatial events in the manner of DAW-software. The X-axis represents time. Events assigned to each ID are represented as gray rectangles in the view, and the movements of each ID is displayed as solid and dotted white lines in the rectangles. The thumbnails of the corresponding sound paths are rendered on top of them.

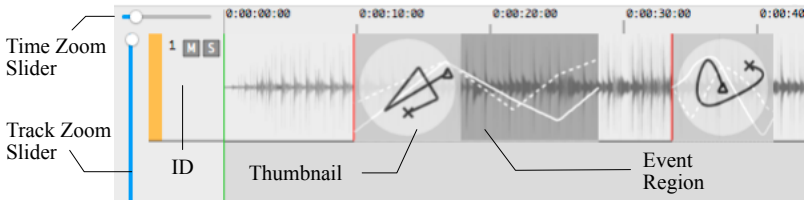


Figure 2.5: Events displayed in Event View

## 2.5 Transport Bar

Transport bar consists of seven buttons, three text fields, the Sync menu and the HRTF button. The functions of each are described below:

Transport_bar.png

Figure 2.6: Transport Bar

#### Rewind Button

set the position of playback cursor to 0:00:00:00.

#### Jump To Previous Marker

Set the current time to the time of the previous marker. If no previous marker is found, it jumps to 0:00:00:00

#### Jump To Playback Start Time

Set the current time to the start time of the previous playback. This is useful if you want to listen to the previously played part once again. In the event view, this position is indicated with a vertical dark green bar.

#### Play/Pause Button

When clicked, Zirkonium starts to play the piece back from the position where the playback cursor is located. If clicked again, the playback is paused.

#### Record Button

When this button is turned on, it allows for real-time drawing of trajectories with the mouse on the Dome View while playback is on, automatically creating an event.

#### Jump To Next Marker

Set the current time to the position of the next marker.

#### Loop

If activated, the playback is looped between the loop start time and the loop end time.

#### Playback Cursor Position

This field indicates the position of the playback cursor. This field is editable and you can input a time that you would like to jump to.

#### Loop Start Time

This field sets and displays the start time of the loop.

#### Set Loop Start Time with Playback Cursor Position

Copy the current time indicated in the Playback Cursor Position field to the Loop Start Time field.

#### Loop End Time

This field sets and displays the end time of the loop.

#### Set Loop End Time with Playback Cursor Position

Copy the current time indicated in the Playback Cursor Position field to the Loop End Time field.

#### Piece Duration

This field shows the duration of the piece. Zirkonium determines the duration of the piece automatically by comparing the end time of the last event and the duration of the longest sound file, and adopts the longer one as the piece duration. This field is not manually editable.

#### HRTF Button

This button with a headphones icon is called **HRTF button**. When enabled, all audio signals are fed to the HRTF possessors, and simulate virtual 3D audio for headphone listening. The processed sound is sent to channel 1 and 2.

**Sync mode pop-up**
With this pop-up menu, you can select synchronization mode. Refer to section 9.4 of chapter 9 for details.

# 3. Setting up

## 3.1 Setting up loudspeakers

The first thing that needs to be done after launching the software is to load the position of each loudspeaker in space.

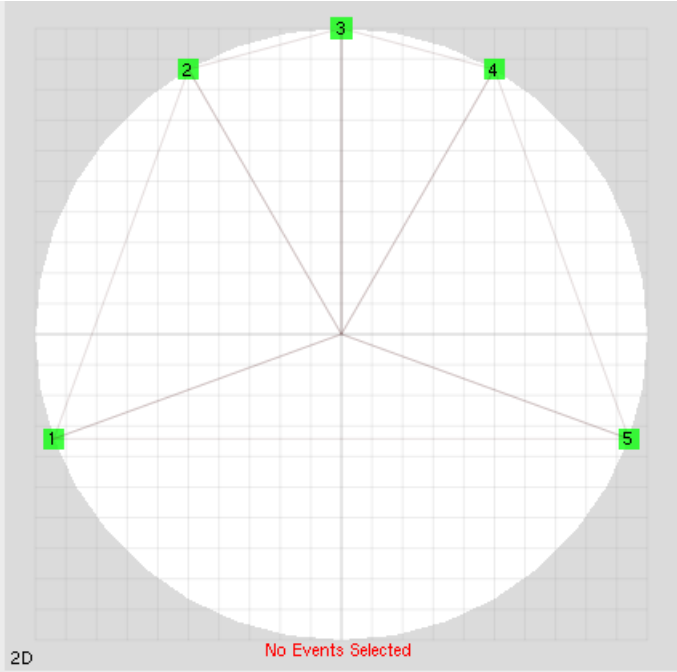


Figure 3.1: 5.0 Speaker Setup loaded on Dome View

### 3.1.1 Standardized Speaker Setup

To set up a standardized loudspeaker configuration, such as stereo, quadraphonic, or 5.0 [fig:3.1], simply select a speaker setup from **File -> Load Speaker Setup**.

### 3.1.2 Custom Speaker Setup

If you would like to use a speaker configuration not listed in the Load Speaker Setup menu. Use the **Speaker Setup** Application for configuring your own setup and exporting the configuration as a XML file. For details, refer the chapter 11 of this user guide.

By selecting “Load from XML File...” under the “Load Speaker Setup” menu in “File”, you can load your custom speaker configuration to Dome view.

## 3.2 Audio Settings

The detailed properties for each input channel can be configured with the **Audio Settings** sheet. To open this sheet, select **File -> Audio Settings**.

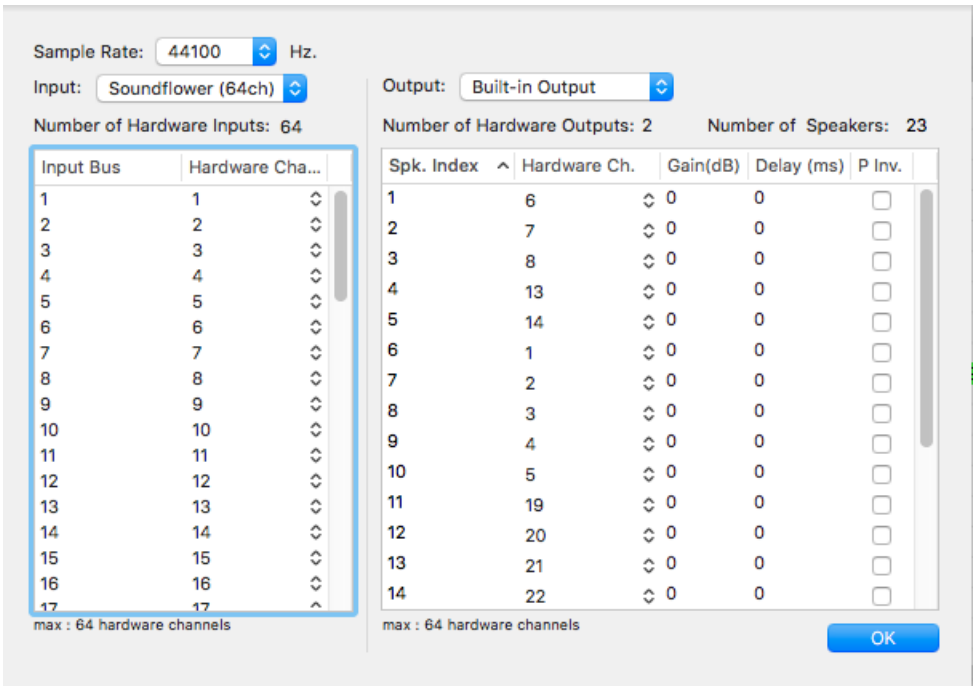


Figure 3.2: Audio Setting Panel

Following, is a description of each item in the sheet.

### 3.2.1 Sample Rate Pop-up

Select sample rate from **44100**, **48000** or **96000** Hz, using this pop-up menu. It is highly recommended to setup the sample rate before importing sound files to the software.

### 3.2.2 Input Device Pop-up

Select a hardware device for audio input from the pop-up menu. You can receive audio signals not only from physical devices connected to the computer but also other software running on the computer by installing Soundflower or similar utilities.

### 3.2.3 Output Device Pop-up

Select a hardware device for audio output from the pop-up menu

### 3.2.4 Input Patch Table

This table determines the relationship between the Hardware inputs and virtual input buses. This is useful when your hardware offers a large number of inputs and you want to use, for example, channel 128 of your hardware.

The input buses determined in this table are used as the input channels when you select "Live Input" as the source of an ID.

**Input Bus** The input bus to which the selected Hardware channel is connected

**Hardware Channel Pop-up button** The hardware channel that is connected to the input bus

### 3.2.5 Speaker Property Table

A speaker setup must be loaded to edit this table, in it, you can configure the following parameters for each speaker:

**Index** The index of speaker as defined in the Speaker Setup XML File. This parameter is not editable.

**Hardware Channel** Output channel of the audio hardware (e.g. your audio interface) assigned to each speaker.

**Gain** The gain of the speaker in dB. 0 means Unity (bypass). The range of this parameter is between -30 and +10.

**Delay** Delay time applied to the speaker. This parameter is useful for avoiding comb-filter effects caused by irregular speaker setups.

**Phase inversion** If checked, Zirkonium inverts the signal output of the specified speaker.

# 4. Importing sources

## 4.1 What is a Source?

In Zirkonium, the term **source** represents all kinds of audio data that can be utilized for spatialization. A source can be in form of a sound file, such as “my sound.wav” stored in the hard drive, but it can also be a live signal coming directly from an audio interface, attached to your computer.

## 4.2 Sources Table

The source table lists all available sound sources. Each column of the table describes a property of the sources [fig:4.1].

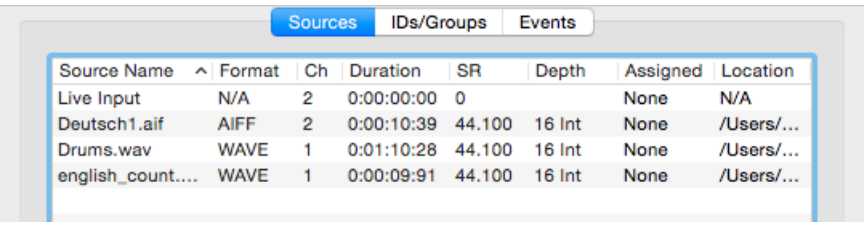


Figure 4.1: Sources Table

### Source Name

The name of the imported audio file

### Format

The format of the imported audio file. WAVE or AIFF

### Ch

The number of audio channels

#### Duration
The duration of the audio file in *h:mm:ss:cs* format

#### SR
The Sample Rate of the file

#### Depth
The Bit Depth of the file (Usually 16, 24 or 32 bits)

#### Assigned
IDs that currently use this file

#### Location
The full path of the file

R **Live Input** represents the audio input from your audio interface or microphones attached to or embedded in your computer. This entry is not removable from the source list.
In case you use software that transfers audio signal from one application to another application such as SoundFlower, select "Live Input" for receiving the audio signal from said software.

R When sound files are imported, Zirkonium creates their waveform images. This may take several seconds, depending on the sample rate and the duration of the files.

## 4.3 Adding and Removing Files

To add sources to the project, simply drag and drop sound files onto the “Sources” table, or click the “+” button below the table and select files from the open panel. Multiple files can be added at the same time.

To remove source file(s) from the sources table, simply select the files you wish to delete and press backspace.

## 4.4 Acceptable File Formats

Zirkonium currently only accepts sound files that have the following properties:

#### Number of Channels
- 1 - 8

#### File Format
- AIFF
- WAV

#### Sample Rate
- 44.1 kHz
- 48 kHz
- 96 kHz

#### Bit Depth
- 16 bit
- 24 bit
- 32 bit

The sample rate of the file must conform to the sample rate of the project. If the sample rate of the file does not match the sample rate of the project, these files are not assignable to IDs.

To change the sample rate of the project refer to chapter 3.2.

> You can remove a sound file assigned to an ID from the sources table. If you do this, the associated IDs lose the connection to sound file, but it remains in the IDs list. The events associated to an ID are also retained in the event table for later usage.

# 5. Creating IDs

## 5.1 What is an ID?

**ID** refers to a sound object in Zirkonium. Before a sound source can be moved in space, you need to create an ID and assign a source (sound file or live input) to it.

> In Zirkonium, the number of channels that are assignable to single ID is always **one** (i.e. mono). If you want to move a stereo source in a space, import a stereo file, create two IDs, assign each ID to each channel of the imported stereo file, and group them together using the grouping functionality described in chapter 6.

## 5.2 IDs Table

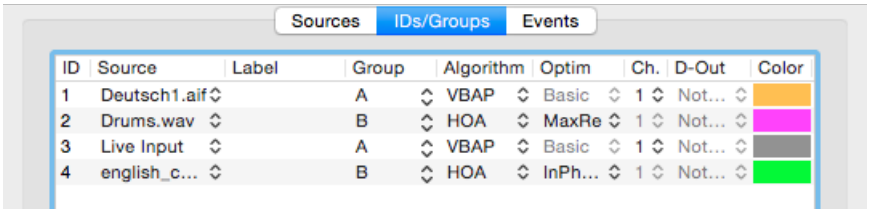


Figure 5.1: IDs Table

The **IDs table** ,located at the top of the IDs / Groups tab[fig:5.1], shows all IDs in the project and displays the properties of each ID listed below:

### ID

The index of the ID assigned automatically by the software. This value is not editable.

### Source

The audio source (sound file or live input) assigned to the ID.

#### Label
The name of the ID.

#### Group
The group that the ID belongs to.

#### Algorithm
The spatial rendering algorithm assigned to the ID.

**VBAP** Vector Based Amplitude Panning (Default).
**HOA** High Order Ambisonics.
**OSC** Open Sound Control. If this option is selected, Zirkonium does not process the audio signal and simply sends the position of the ID as OSC messages. This option is useful if you want to control another software or hardware with Zirkonium.
**None** Doesn't output any audio signals or OSC messages from this ID, unless D-out (Direct Out) is activated.

#### Optim
Optimization option for HOA Spatial rendering algorithm.

When HOA (High Order Ambisonics) is selected as the spatial rendering algorithm, an optimization option for the algorithm becomes selectable and you can choose one of the following optimization options. The definition of the three optimization options by the CICM (developer of HOA library) are as follows:

**Basic** has no effect, it should be used (or not) with a perfect Ambisonics channels arrangement where all the channels are to equal distance on a circle or a sphere, and for a listener placed at the perfect center of the circle of the sphere
**MaxRe** should be used for an auditory confined to the center of the circle of the sphere.
**In Phase** should be used when the auditory covers the entire channels area and when the channels arrangement is not a perfect circle or a perfect sphere or when the channels are not to equal distance.

> The order of Ambisonics is optimized automatically by the software, based on the number of loudspeakers; higher number of loudspeakers requires higher Ambisonics order. Thus, CPU consumption by HOA may increase with a higher number of loudspeakers.

#### Ch
The channel assigned to the ID. When the source is mono, the pop-up is disabled.

#### D-Out
Direct Out. When assigned, the source signal will be routed directly to the selected output of the audio hardware.

> D-Out is useful when you want to send a sound source to a single specific speaker or send it to another external processor.

#### Color
The color that represents each ID. When this cell is clicked, a color picker floating window should appear and let you choose the new color for the ID.

## 5.3 Creating a new ID

To add an ID to Trajectory Editor, first, while in the “ID/Groups” tab, click the “+” button below the upper table to add a new ID.

When an ID is added to the IDs table, an **initial event** is created and the position of the ID is set to ( 0, 0 ).

> This initial event is not removable from the event table manually and most of the properties of the initial event are not modifiable by the user except **initial position**. An initial event is automatically removed when the corresponding ID is removed from the IDs table.

## 5.4 Assign a source file to an ID

Simply click on the pop-up menu and select the source you want to assign to the ID.

## 5.5 Deleting ID(s)

To delete IDs, Select one or multiple IDs listed in the IDs table and press the backspace key.

Caution - all spatial events associated with the deleted ID will also be deleted from the event list.

# 6. Bundling as Groups

## 6.1 What is a Group?

**Group** represents a set of multiple IDs moving together. Grouping function allows us to move several IDs along a single sound path. All IDs, belonging to a Group, are called **member IDs** of a group. A group must have a single **Master ID**, all the other IDs in that group are called **Slave IDs**. The Master ID moves along the sound path and each slave ID follows the master ID, keeping the distance or angle between them [fig:grouping].

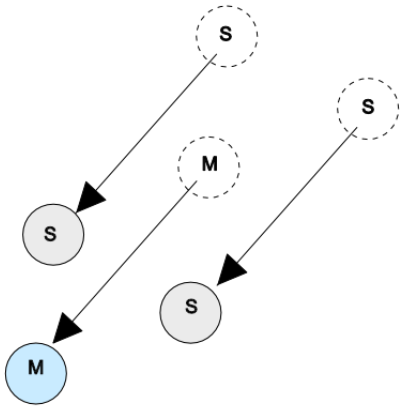


Figure 6.1: Grouping

R You are only able to draw a sound path for a group, if a master ID is assigned. The Master ID can be selected in the Groups table.

## 6.2 Groups Table

The Groups table shows detailed properties of each group.

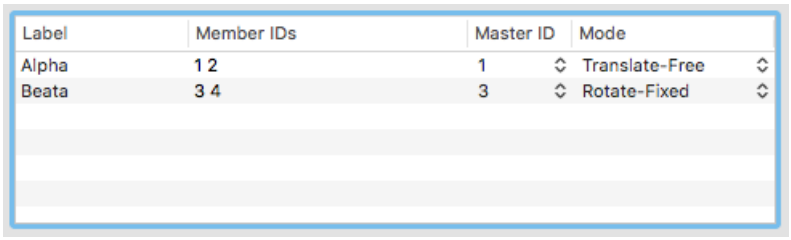


Figure 6.2: Groups Table

#### Label

The name of each group. It is highly encouraged to provide each group with a proper name.

#### Member IDs

This column displays IDs that belong to the group. Not editable.

#### Master ID

The leading ID of the group

#### Mode

The mode of the group. This mode determines how Slaves follow the Master ID. This is discussed in 6.11 in detail.

**Translate-Fixed** During a group event, the Cartesian distance between the master ID and all slave IDs is determined by the Cartesian distance of the initial position.

**Rotate-Fixed** During a group event, the spherical distance between the master ID and all slave IDs is determined by the spherical distance of the initial position.

**Translate-Free** Same as Translate-Fixed, but in this mode, the relations between positions will adapt to a new arrangement caused by individual events of the member IDs.

**Rotate-Free** Same as Rotate-Fixed, but in this mode, the relations between positions will adapt to a new arrangement caused by individual events of the member IDs.

**Mirror** The number of members of this mode must be two. The position of the slave is always equal to the position of the master, mirrored around the Y axis.

## 6.3 Creating a Group

To create a group, click the “+” button below Groups table. Then, a single group should appear in the table.

## 6.4 Naming a group

To name a group, double click on the label cell of a group and type the name in. The name should be settled when the return key is pressed or when you click somewhere else.

## 6.5 Adding a member to a group

To add an ID to a group, select the desired group from the group column of the ID’s table.

## 6.6 Removing a member from a group

To remove an ID from a group, select another group or "unset" in the IDs table. Note, you can not remove the master ID from a group. If you want to remove the master ID from a group, change the Master ID in the group table first, then remove it.

## 6.7 Setting a Master ID

To select the master ID of a group, click on master ID pop-up cell in the Groups table, and select an ID from the member IDs. A sound path for a group cannot be drawn, unless a master ID is assigned.

## 6.8 Selecting the group Mode

To select a group mode, click the pop-up menu and select one of the available five modes. The difference between the five modes is described below.

## 6.9 Changing the group an ID belongs to

To change the group that an ID belongs to, simply select another group in the ID table. To remove an ID from a group, select “unset” from the pop-up.

## 6.10 Deleting a Group

To delete a group, select a group or groups in the list and press the backspace key.

> When you delete a group. All events that control the deleted group will lose their target but will remain in the event table. You can then associate these events with another ID or Group.

## 6.11 Which group mode to select

You can select a group event mode from 5 modes.

### 6.11.1 Translate and Rotate

In the translate modes (Translate-Fixed or Translate-Free), the Cartesian distance between each ID and master ID is kept unchanged and the size of group is constant. However, the relationship between IDs and the listener varies, depending on the position of the group. As the figure 6.3 shows, the group creates a stereo image if it is placed in front of the listener, but the distance between the ID’s becomes indistinguishable if it is moved to the left or right side of the listener.

In contrast, in the rotate modes (Rotate-Fixed or Rotate-Free), the spherical distance (or angle) between each ID and Master ID is kept unchanged. You can take full advantage of this mode if you move the group in a circle manner (rotating around the center).

This mode causes two possibly unwanted side effects. Firstly, in this mode, the Cartesian distance or the size of a group is changed, depending on the elevation. If a group moves towards the zenith, the distance between IDs is reduced, and if a group moves towards the horizon, the distance between IDs expands. Secondly, the formation of a group will be mirrored if the group travels through the zenith.

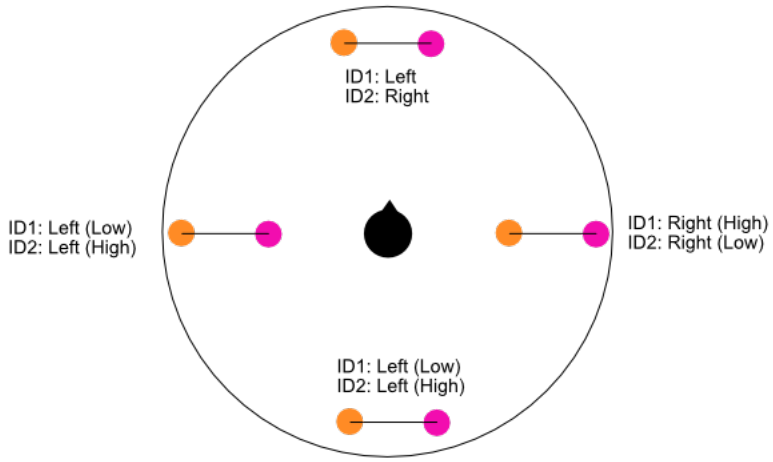


Figure 6.3: Translate Mode

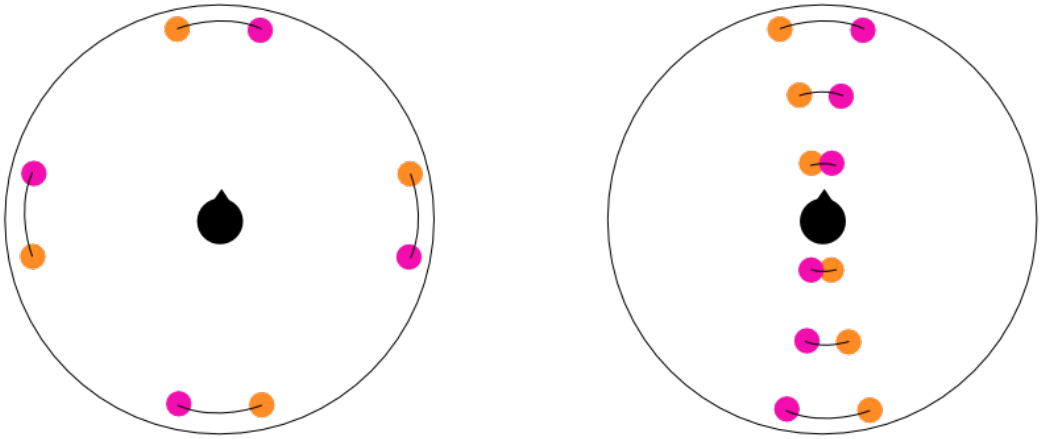

Figure 6.4: Rotate Mode

### 6.11.2 Fixed and Free

Fixed mode means the distance between the master ID and each slave ID is determined by, and only by the initial position of the IDs. This means that the initial formation cannot be changed, if any ID moves independently, after said movement, the formation is reset to the initial one. This mode is useful if you want to keep the formation of a group unchanged throughout a piece.

In Free mode, the distance between master and each slave ID is determined by the final position of latest individual single ID event. Meaning that the initial formation adapts to any individual ID movements that may occur, retaining the most recent formation. This mode is useful if you want to change the formation of a group in a piece.

### 6.11.3 Mirror mode

The Mirror mode is a special mode and different from the other four modes. The number of members in a mirror-mode group must be 2. The position of the slave ID is always mirrored around the Y axis. If the position of the master is (x, y) the position of slave is always (-x, y). Because of

this exceptional behavior of the slave, there is no option for fixed or free mode (Fig. 6.5).

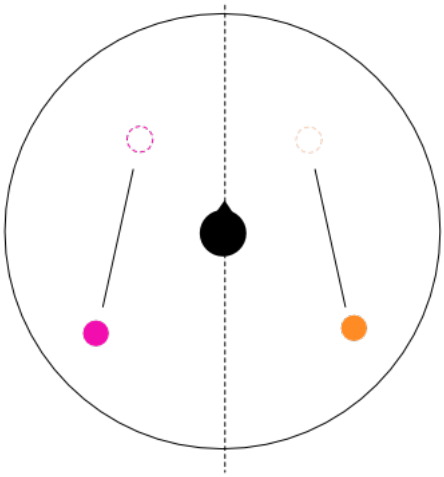

Figure 6.5: Mirror Mode

The mirror mode is useful when you want to keep the stereo image of the sound sources always balanced.

# 7. Managing events

## 7.1 What is an Event

In Zirkonium, all spatial movements of IDs and Groups are called **Events**. Usually an event has a start time and an end time.

An event has several properties that are related to either time or space. To edit the time-related-properties, such as start time and end time, we use **event view** or **table view**. To edit the space-related-properties, such as sound path or motion path, we use **dome view** and **motion view**.

There are two special types of events; **initial event** and **marker event**. Initial events determine the initial position of an ID. A marker event represents a certain time point of the piece, for example the beginning of a section, and doesn't possess spatial properties.

## 7.2 GUI Overview

### 7.2.1 Event Table

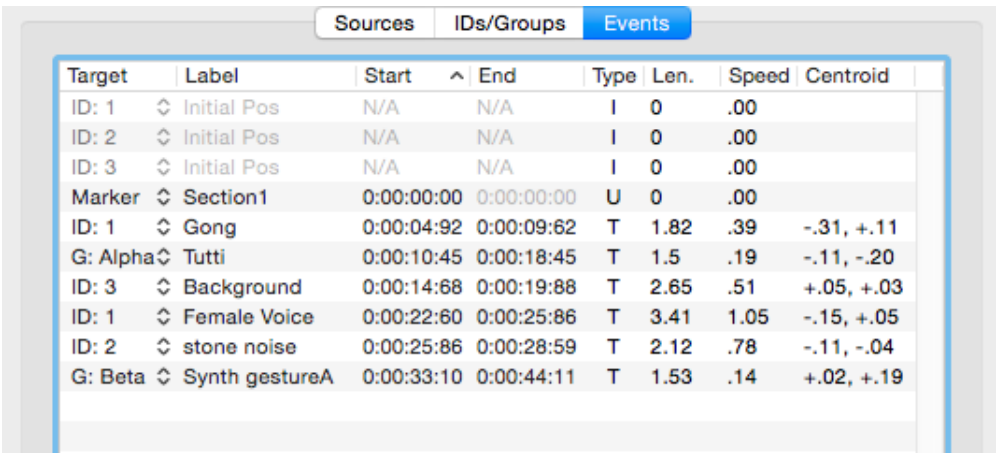


Figure 7.1: Event Table

Click the "Event" tab to display the event table. In the event table, all events are listed chronologically. You can directly modify some of the properties in the table. The descriptions of the properties are as follows.

### Target
You can select the target of each event from marker, IDs and Groups in the pop-up menu. The default setting is "No Value". The dome view and motion view are disabled unless you select an ID or a Group target.

### Label
The name of each event. This label will be displayed in the dome view, if the visibility of the event name is enabled (View - Dome View - Sound Path - Event Name).

### Start Time
The start time of the event in *h:mm:ss:cs* format. You cannot set a time after the end time of the piece.

### End Time
The end time of the event in *h:mm:ss:cs* format. You cannot set a time before 0:00:00:00. This cell is disabled the event is a Marker.

### Type
The type of each event indicated with a letter, U, I or T.

**undefined (U)** An undefined event has no anchor points. The event doesn't move the position of IDs at all.

**instantaneous (I)** An instantaneous event has only one anchor point and the ID moves to the position of the anchor point instantly when the play cursor reaches the start position of the event.

**trajectory (T)** A trajectory event has more than one anchor point in its sound path and the ID moves, in principle, from the start to end position.

### Len.
The length of the path measured in relation to the radius, where the radius of the unit circle = 1.0.

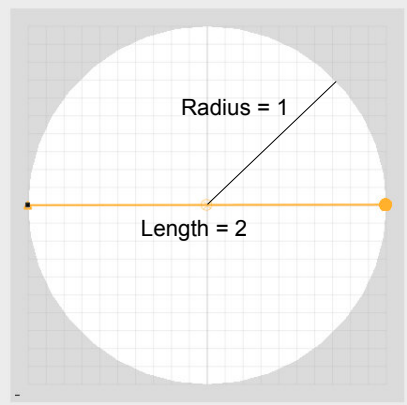


Figure 7.2: Length of a sound path

### Speed
The average speed of the trajectory.

### Centroid

Zirkonium internally converts (rasterizes) the sound path drawn with Bézier curves to a large number of points, in order to optimize playback. At the same time, it calculates an average coordinate of these points. This average coordinate is called **Centroid** and it indicates the approximate spatial center of the sound path.

## 7.2.2 Dome View

The Dome View is the main canvas of Zirkonium. Here you can draw sound paths freely with Bézier curves.
The most important GUI elements of the Dome View are listed below.

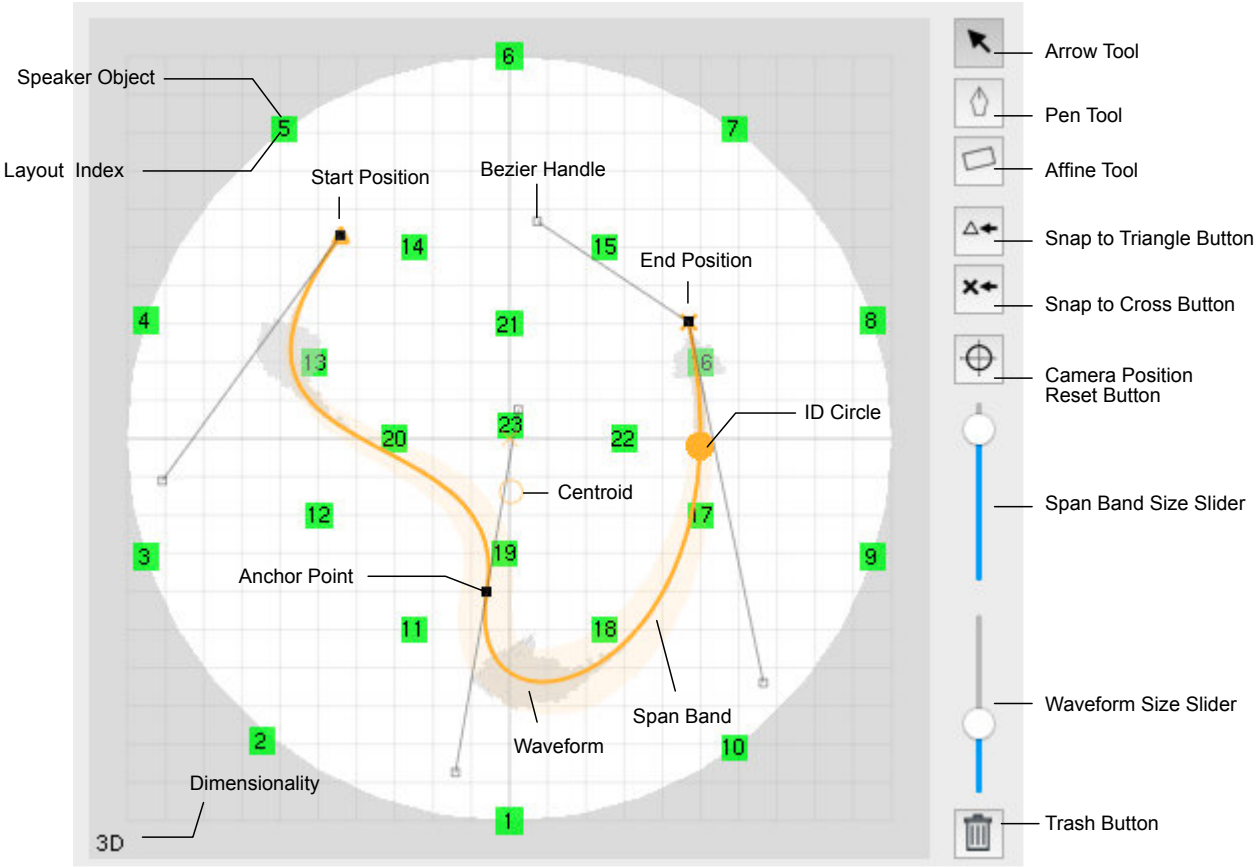


Figure 7.3: Dome View

### Speaker Object

The position of each Speaker is indicated as a green square and the layout indices defined in the loaded Speaker Setup XML file are shown in it.

### Dimensionality

This shows the dimensionality of the current speaker setup. The dimensionality is defined in the Speaker Setup XML file and is not modifiable in Trajectory Editor. The CPU resource consumption significantly increases if the loaded setup is 3D.

### Sound Path

The actual path that an ID moves along. The color of the sound path is determined by the color of the ID. The following are graphical symbols attached to a sound path:

**Start / End Position** The triangle symbol indicates the start position of a sound path, and the cross symbol indicates the end position. At the start time of an event, an ID starts to move from the start position, and it reaches the end position at the end time of the event.

**ID Circle** This circle indicates where the ID is currently located along the sound path.

**Centroid** The geometrical center of the trajectory.

**Waveform** Representation in space of the audio in the associated source file.

**Span Band** The size or expansion of the ID represented as a band.

**Anchor Point** Represented as solid black squares, these are "joints" of multiple Bézier curves.

**Bézier Handles** These handles are used to control the arch of each Bézier curve.

#### Arrow Tool

When active, you can select items in the dome view, motion view and event view.

#### Pen Tool

When active, you can add anchor points in the dome view or motion view and create event regions in the event view.

You can switch between Arrow Tool and Pen Tool quickly by the keyboard short-cut: Cmd+E.

#### Affine Tool

When selected, you are able to translate, rotate and scale the sound path in the dome view.

#### Snap To Cross Button

The start position of the currently selected sound path will be matched to the transparent cross (i.e. the end position of the sound path of the previous event). By matching these two points, you can avoid abrupt jumps of the ID position.

#### Snap To Triangle Button

The end position of the currently selected sound path will be matched to the transparent triangle (i.e. the start position of the sound path of the next event). By matching these two points, you can avoid abrupt jumps of the ID position.

#### Center View Button

Center the Dome View.

#### Span Band Size Slider

This slider controls the size of the span band displayed in the dome view. Note: this slider modifies only the visual representation of the span and doesn't change the actual span path.

#### Waveform Size Slider

This slider controls the size of the waveform attached to the sound path. Note: this slider modifies only the visual representation of the waveform, and doesn't change the actual amplitude of the audio.

#### Trash Button

When clicked, the sound path shown in the dome view, will be deleted instantly.

#### Visibility Control

The visibility of each GUI component is configurable with the Dome View menu under the View menu. The Dome View menu has three sub-menus: Plane, Speaker and Sound Path. The details of each menu items are listed in the tables 7.1 - 7.3.

| | |
|---|---|
| Grid | show/hide the 10x10 grid in the dome view |

Table 7.1: Plane Sub Menu

| | |
|---|---|
| Object | show/hide the speakers (depicted as green squares) |
| Index | show/hide the layout index number of each speaker |
| Output | show/hide the hardware output number associated with each speaker |
| Ring | show/hide the rings (speaker groups) |
| Triplet | show/hide the loudspeaker triplets and currently selected triplet used in VBAP algorithm |
| Gain | show/hide the gain of each speaker |
| Level | if activated, the output level of each speaker is color-coded from green to red |

Table 7.2: Speaker Sub Menu

| | |
|---|---|
| Waveform | show/hide the waveform displayed along the sound path |
| Ruler | show/hide the ruler |
| Centroid | show/hide the geometrical center of the sound path |
| Event Name | show/hide the name (label) of the event that the sound path belongs to |

Table 7.3: Sound Path Sub Menu

### 7.2.3 Motion View

The motion view is an editable line graph that represents the relationship between time (X-axis) and the relative position of an ID along a sound path (Y-axis).

As written above, a sound path has a start position and an end position, which are marked with a triangle and a cross symbols respectively. The vertical line on the left edge of the view is called Mini-Sound Path and it represents the tightened up version of a sound path, drawn in the Dome View.

The descriptions of graphical components in Motion View are as follows:

#### Start Time

The start time of selected event.

#### Duration

The duration of selected event.

#### End Time

The end time of selected event.

#### Mini-Sound Path

This is a compact representation of a sound path drawn in the dome view. The start position of a sound path corresponds with the bottom end, and the end position corresponds with the top end of the Mini-Sound path. The Mini-Sound Path also displays the relative position of anchor points on the sound path.

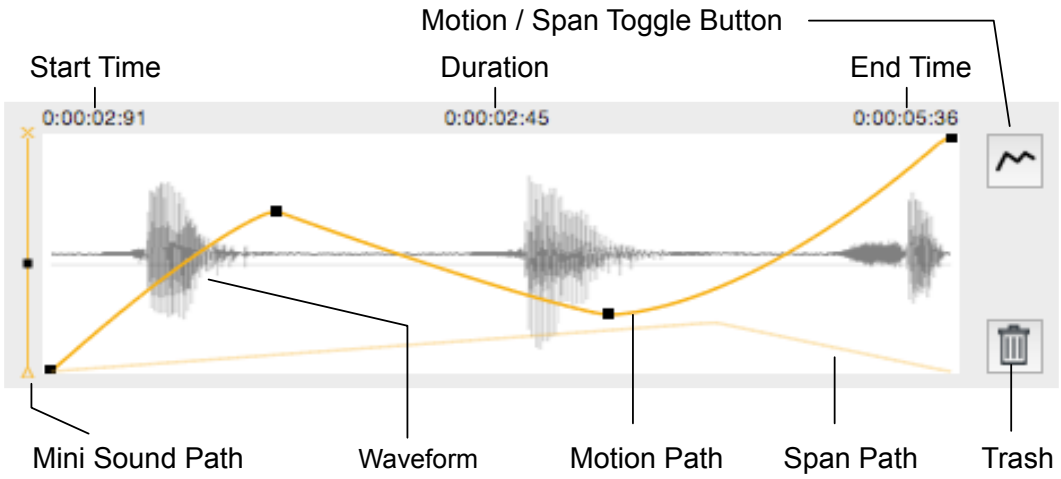


Figure 7.4: Motion View

#### Motion and Span Path

If the motion mode is selected, the view represents position in the sound path in the Y axis. If not, the span path is displayed, representing span size in the Y axis. These modes can be alternated by pressing Motion / Span toggle button.

#### Waveform

The waveform of an audio content to be played during the selected event.

#### Motion Span Toggle Button

You can toggle motion and span mode by clicking this toggle button.

#### Trash Button

When clicked, both motion and span path will be reset.

#### Visibility Control

The visibility of each GUI component is configurable with the Motion View menu under the View menu.

| | |
|---|---|
| Waveform | show/hide the waveform displayed behind the motion / span path |
| Ruler | show/hide the ruler |
| Anchor Points as Lines | show/hide horizontal lines from the anchor points on the Mini-Span Path |

Table 7.4: Motion View Visibility Control

"Anchor points as Lines" function is useful to adjust a certain audio content and a position on a motion or span path. For details refer to 7.6.2.

### 7.2.4 Event View

The Event view visualizes the created events graphically in the manner of DAW-Software. You can edit the time-related properties, such as start time or end time of events directly here in the event view. The detailed description of each graphical component are as follows:

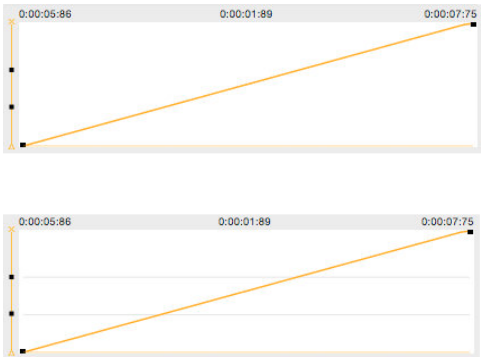


Figure 7.5: Anchor Points as Lines disabled and enabled

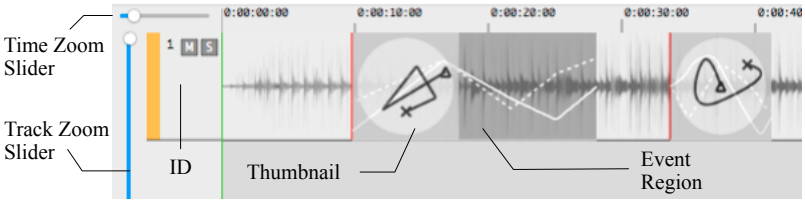


Figure 7.6: Event View

#### ID

Each track represents the audio content and the movements of an ID. The number of each ID is displayed next to the ID color bar.

#### Mute / Solo Button

Like in a DAW software, you can temporarily mute a specific ID by clicking “M” (Mute) button or you can mute all other tracks by clicking “S” (Solo) button.

#### Time Zoom Slider

This slider controls the horizontal zoom factor of the event view.

#### Track Zoom Slider

This slider controls the vertical zoom factor of the event view.

#### Waveform

Actual audio content of the associated source file.

#### Event Region

In the event view, events are represented as dark rectangles called **event regions**. On top of each event region, a thumbnail, XY-Movement Lines and a Jump Warning line are displayed.

**Thumbnail** A small snapshot of the sound path

**XY Movement - Span Lines** The solid white line displays the movement of ID on X axis, the dotted white line shows the movement of ID on Y axis. The Grey solid line indicates the change in the span.

**Jump Warning Line** The red line at the left edge of each event region warns you that the listener might perceive an unnatural abrupt sound movement (i.e. jump) at the moment of the red line, due to a sudden change of the ID position.

#### Visibility control

The visibility of each GUI component is configurable with the Event View menu under the View menu.

| | |
|---|---|
| SoundPath | show/hide lines that indicate the movement of ID on X and Y axis |
| Thumbnail | show/hide thumbnails of sound paths |

Table 7.5: Event View Visibility Control

## 7.3 Navigating the playback cursor

Zirkonium offers several ways to navigate the playback cursor.

### 7.3.1 Playback Cursor Position Field

The first option would be the Playback Cursor Position field. Simply set a time in *h:mm:ss:cs* format in the field. The green playback cursor will then jump to the provided time instantly.


Figure 7.7: Navigating playback cursor with the playback cursor position field

### 7.3.2 Markers

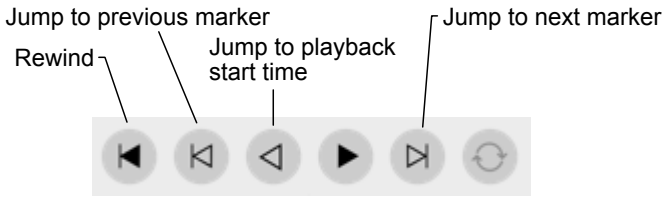


Figure 7.8: Navigating playback cursor with the transport bar

You can easily jump to a specific point in time, using the “jump to previous marker” or “jump to next marker” buttons in the transport bar.

The rewind button instantly set the playback cursor position to 0:00:00:00.

### 7.3.3 Jump to playback Start Time

When you start playback, Zirkonium automatically stores the time point that the playback began. This time point is indicated with a vertical dark green line in the event view (Fig. 7.9).

After the playback, if you want to hear once again the same part of your piece. Simply click the Jump to playback start time button (or Cmd+R). The playback cursor will then jump immediately to the position where the previous playback started.

### 7.3.4 Loop Playback

Zirkonium Trajectory editor also offers a loop playback function, if you want to listen to a specific part of your piece repeatedly. In order to specify the loop start and end time, use the loop start / end field under the Playback Cursor Position field.


Figure 7.9: Playback start time is indicated with a thin dark green line in the Event View

You can enter the loop start / end time by typing time in *h:mm:ss:cs* format, or, use the Set Loop Start/End Time with Playback Cusror Position button to copy the current time to the field.

The loop start and end times are indicated with red lines in the event view.

Figure 7.10: The transport bar with the activated loop button

Then, activate the loop mode by clicking the Loop button and press the Play button. To deactivate the loop mode, click the Loop button again.

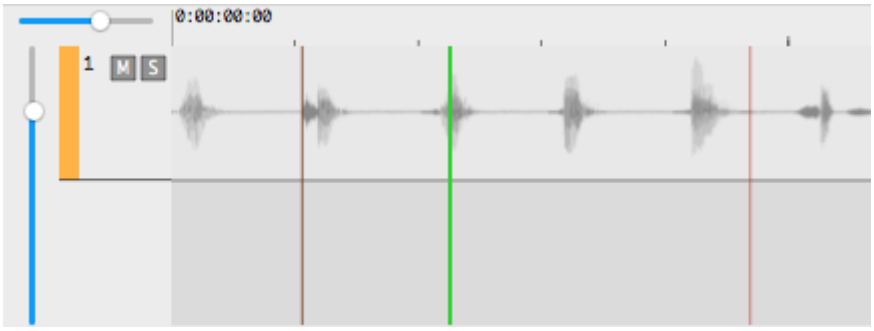


Figure 7.11: The loop start/end time are displayed with thin red vertical lines

### 7.3.5 Time Ruler

There is a time ruler above the event view. You can click on the time ruler to adjust the position of playback cursor. By double clicking on the time ruler, you can also start playback from the clicked position immediately.

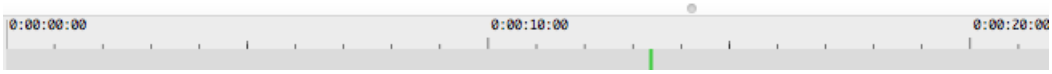


Figure 7.12: Using Time Ruler for the navigation

## 7.4 Creating an Event

| Target | Label | Start | End | Type | Len. | Speed | Centroid |
|---|---|---|---|---|---|---|---|
| ID: 1 | Initial Pos | N/A | N/A | I | 0 | .00 | |
| Marker | Section1 | 0:00:00:00 | 0:00:10:00 | U | 0 | .00 | |
| ID: 1 | | 0:00:00:00 | 0:00:20:00 | T | 2.34 | .12 | -.21, -.02 |

Figure 7.13: Three types of events listed in the event table

As mentioned above, Zirkonium handles three types of events: normal event, initial event and marker event. You need to follow different steps to create these three different types of events.

### 7.4.1 Normal Event

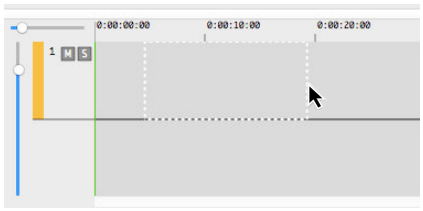


Figure 7.14: Creating an event with the pen tool in the event view

There are two ways to create a normal event that controls the movement of an ID or a group. You can create a new event by clicking the “+” button in the event tab of the table view, and select an ID or a group as a target, using the pop-up menu in the Target column. Alternatively, you can select the pen tool and draw a region in the event view.

If you create a group event, the “shadow” region appears in the tracks of the member IDs of the group. These shadow regions indicate that the movement of IDs are controlled by a group event. For example, in figure 7.15, ID 1 and 3 are the member IDs of a group “Alpha”, thus, the shadow regions appear in the tracks of ID 1 and 3 above the group event.

You are unable to create an ID event that overlaps shadow regions(i.e. group events).

### 7.4.2 Initial Event

As described in chapter 5, an initial event is created for each ID. This initial event is not manually removable from the event table and a sound path cannot be assignable to an initial event. You can only determine a fixed initial position of an ID, using this event.
Initial events are automatically deleted, when the corresponding IDs are deleted.

### 7.4.3 Marker Event

In order to create a marker event, select the “event” tab in table view and click the “+” button. Then, an event with “no target” appears in the event table. Change the target to “Marker” with the pop-up menu under the column, “target”. The time is automatically set to the cursor position and you can provide the newly created marker with a name.

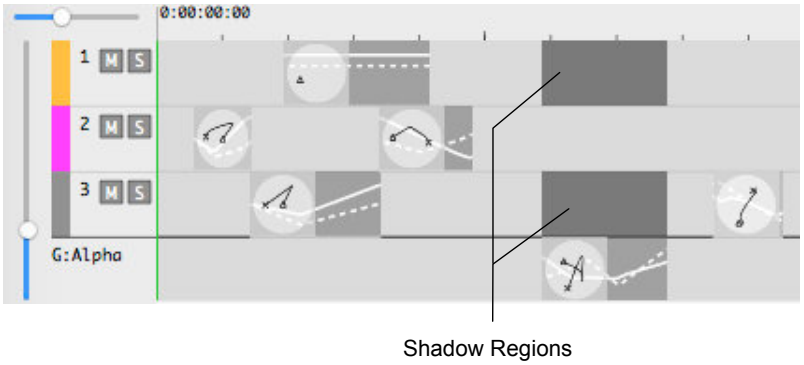


Figure 7.15: Shadow Regions

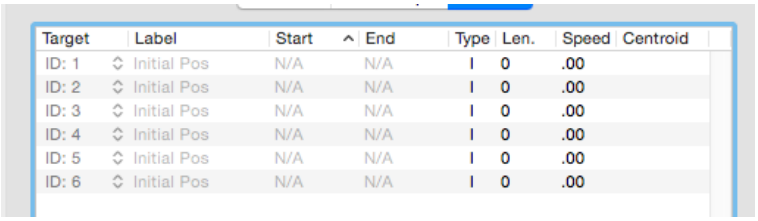


Figure 7.16: Initial Events for each ID

Alternatively, in the event view, you can click the time ruler with the option key pressed. Then, Zirkonium creates a marker event automatically at the clicked position, and names it “newMarker”. The name is modifiable in the event table.

### 7.4.4 Auto Event Creation

To create events for each sound is a tedious work. Zirkonium offers a way to create events automatically based on the amplitude envelope of the loaded audio data.

To use this functionality, load at least one sound file to the source table, create at least one ID in the ID table, and select **(Edit -> Auto-Event Creation)** from the menu, then the Auto Event Creation Sheet will appear on the window (Fig. 7.19).

The Auto Event Creation(AEC) algorithm analyzes the amplitude envelope of the target file by calculating the RMS(Root Means Square) of audio sample blocks. If the algorithm find a block whose RMS value exceeds the provided threshold, the algorithm creates a new event. Then, it attempts to find the end of the event by finding successive blocks whose RMS are continuously below the threshold.

The sheet consists of the following components:

#### Source Selector Pop-up Button

The target source file that the AEC algorithm will analyze. The small pop-up button next to this one is the channel selector.


Figure 7.17: Marker Events can be added directly

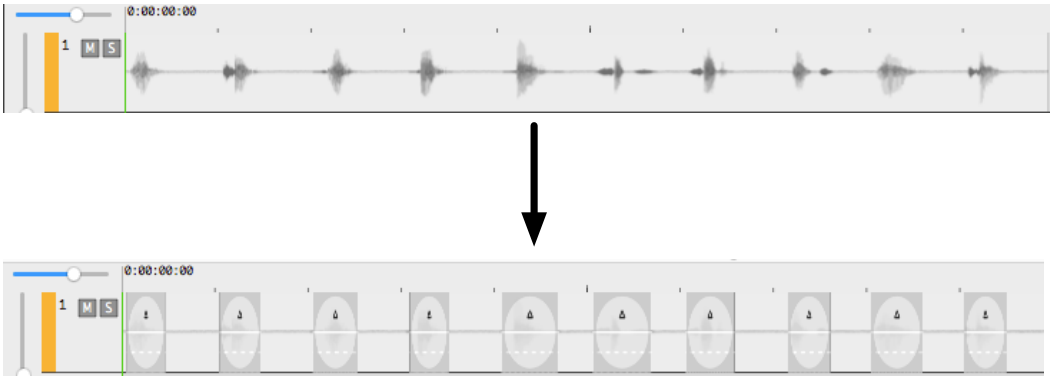


Figure 7.18: Events automatically created by Auto Event Creation

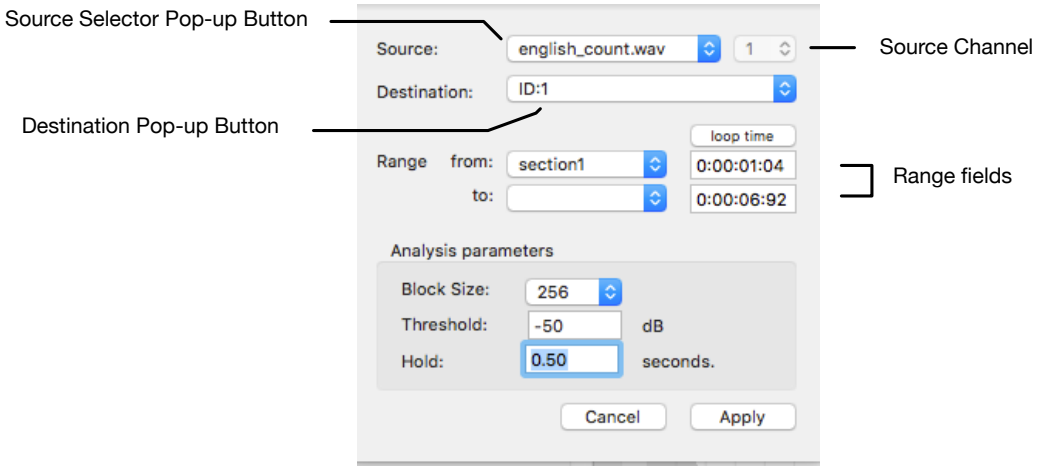


Figure 7.19: Auto Event Creation Sheet

### Destination Pop-up Button

The destination ID or group that the automatically created events will be placed in.

R The destination of the analysis is not limited to its source, this means you can easily create a spatial interaction between two sound sources. For example, when one sound source rings, the other sound source moves. Furthermore, you can select a sound source that is not assigned to an ID, therefore, you can create a series of events based on an audio that the audience will never listen to.

### Range fields

The algorithm only checks the amplitude envelope in the time range defined by "from" and "to" text fields.

You can specify the time range by selecting a marker from the popup buttons or click "loop time" button to apply the loop time defined in the main window.

#### Analysis Parameters

**Block Size** This Popup determine the number of samples for RMS analysis. For a percussive sound, a small block size tends to produce better results.

**Threshold** The threshold parameter for the algorithm in decibel. The default threshold is -50 dB. For a recording with a recognizable background noise, set a higher threshold.

**Hold** The hold value sets the duration of silence that the algorithm requires to determine the end of the event. The algorithm may determine the end of an event based on silence in the sound file. However, for example, a phrase played by a percussion instrument may contain a large number of short silences between each attack. In this case, the algorithm may yield a large number of events. In order to avoid this, set a higher hold parameter. In the figure 7.20 shows the difference of created events resulting from the hold parameter.

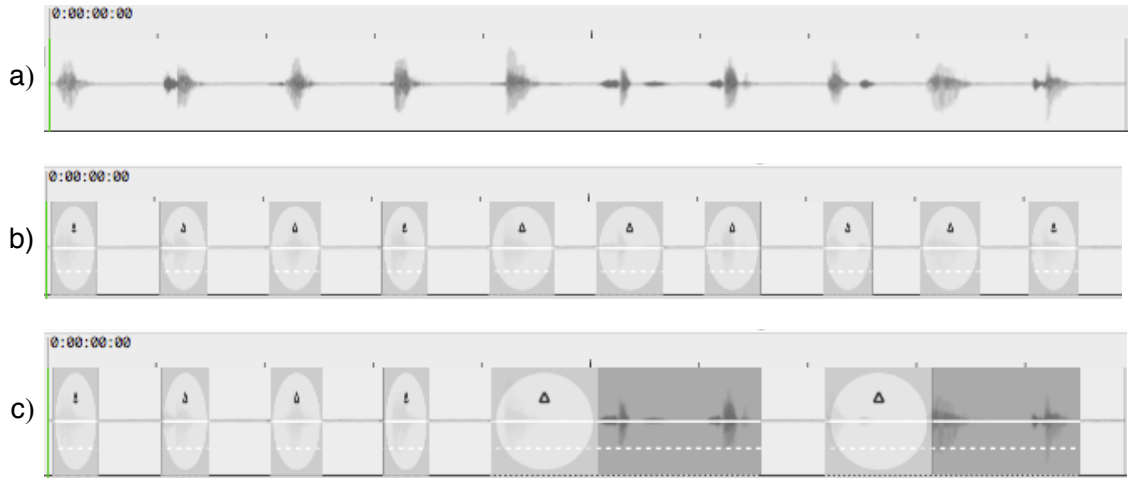


Figure 7.20: Different hold parameters and their results a) sample b) hold = 0.1 sec. c) hold = 0.5 sec.

## 7.5 Editing Time-related Event Properties

You can edit the time-related properties, such as start time or end time in three different interfaces: Event View, Event Table, and Quick Event Manipulator. Zirkonium also provides an event selection panel, which facilitates to select multiple events at once.

R Zirkonium strictly forbids the overlap of multiple events. If you try to create or move event on top of another event, the software alerts you and automatically reverts the modified properties of the event(s).

### 7.5.1 Event View

#### Shifting Events


Figure 7.21: Shift Event

In the event view, you can move the event region by simply dragging the region, using arrow tool.

#### Scaling Events

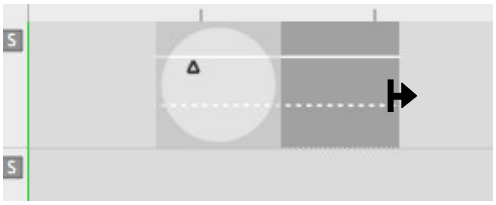


Figure 7.22: Scale Cursor

If you hover your mouse cursor over the left or right edge of an event region, the arrow cursor becomes a scale cursor. This indicates that the region is scalable when you click and drag the edge.

#### Copying Events

To copy events, press the option key and drag the target region(s). It is also possible to shift, scale or copy multiple events simultaneously, simply select multiple events with shift + click. The operation will be canceled if one of the modified events overlaps with other unselected events.

### 7.5.2 Event Table

You can edit the start time and end time of events directly in the event table under the Start and End column by double clicking the cells. The provided time should be positive and conformed to *h:mm:ss:cs* format. If you set the end time later than the piece duration, the piece duration will be automatically updated.

Currently you can name events only in the event table. If you give an event a name, the name will appear in the Dome View, next to the triangle symbol (fig:7.23).

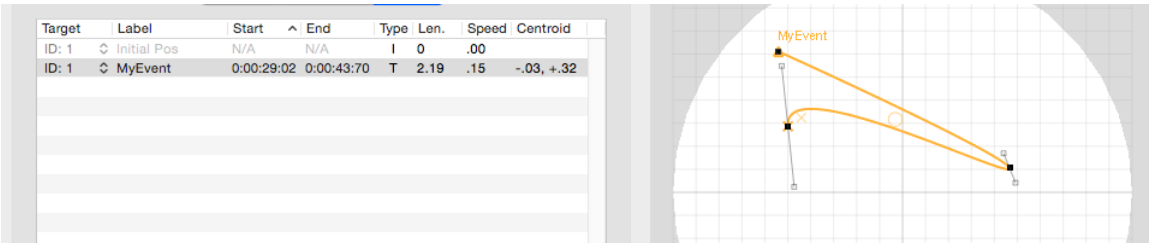


Figure 7.23: Event Name displayed in the dome view

### 7.5.3 Quick Manipulator

Beneath the table view, there is a set of pop-up buttons and text fields. This is called **Quick Event Manipulator**. Whith it, you can shift, scale, or copy multiple selected events precisely.

The Quick Manipulator offers three modes. You can select which of these modes to use with the leftmost pop-up button.

#### Shift mode

In Shift mode, you can shift the start time and/or end time of the selected events. With the shift target pop-up button, you can select the target of the shift operation from start time, end time or both start and end time. Enter the direction of the operation with the direction pop-up, input the amount of shift time in the shift time field and press apply to execute the operation.


Figure 7.24: Shift Mode

#### Scale mode

In Scale mode, you can scale the duration of the selected events. To perform this operation, simply select the direction of scaling, input scale factor in percentage and press the apply button.

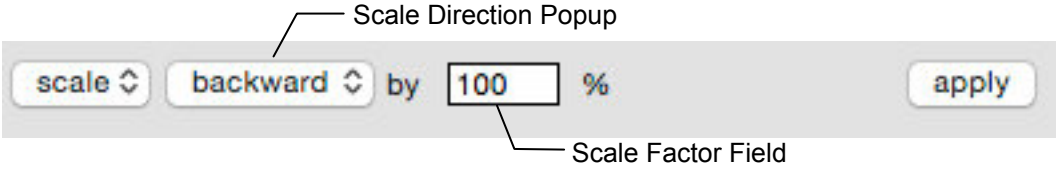


Figure 7.25: Scale Mode

#### Copy mode

In Copy mode, you can copy the selected events to the specified time. Simply enter the target time in the text field and press apply.

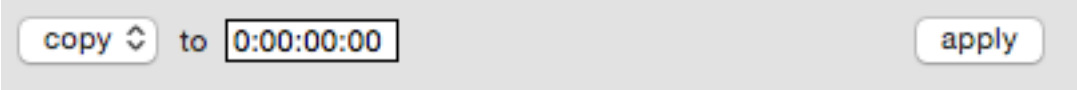


Figure 7.26: Copy Mode

### 7.5.4 Event Selection Sheet

Your project may consist of a large number of spatial events, and it may sometimes be cumbersome to select multiple events you want to modify in event view or event table.

Zirkonium offers a functionality that helps you to select specific events that match certain conditions. In order to activate this functionality, select “Select Events...” from the Edit menu, then, the **Event Selection Sheet** will appear on top of the main window (fig7.27).

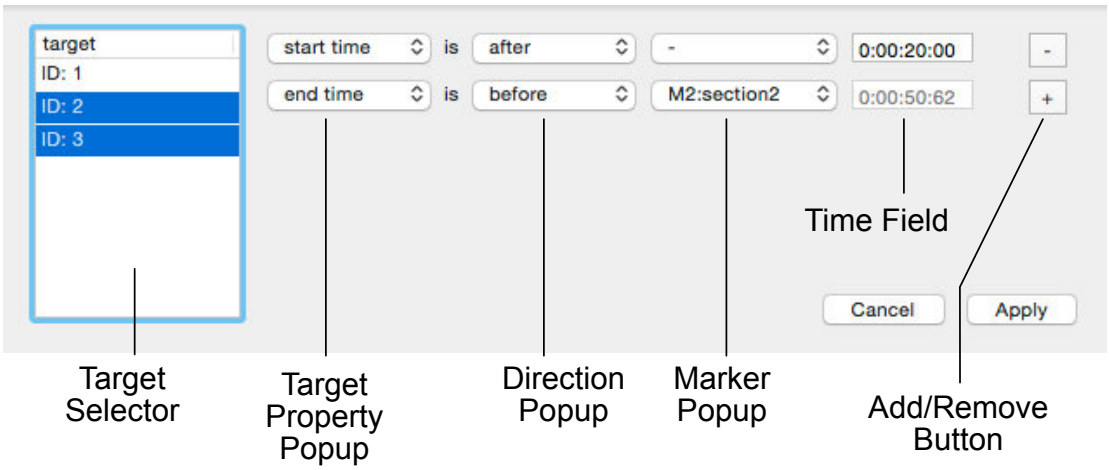


Figure 7.27: Event Selection Sheet

#### Target Selector

Select target IDs or groups for filtering with this selector. You can select multiple targets by shift + click.

#### Target Property Pop-up

Select target property from “start time” and “end time”.

#### Direction Pop-up

Select direction of filtering from “before”, “after”, and “equals to”.

#### Marker Pop-up - Time Field

Enter a time for filtering in *h:mm:ss:cs* format in the time field. Alternatively, select a marker event, using the marker pop-up button. If you select a marker with the pop-up button, the time field will be automatically filled with the start time of the selected marker and will become uneditable.

#### Add/Remove Button

Press these buttons to add or remove a condition.

If all parameters for filtering are set, press the “Apply” button to execute filtering. The sheet will close and all events that match the provided conditions will be automatically selected in the

event table and event view.

## 7.6 Editing Space-Related Event Properties

With **Dome View** and **Motion View**, you can manipulate space-related properties of events that determine the actual position of IDs at a certain moment. In Dome view, you can draw a sound path, a path that an ID moves along. By default, an ID moves from one end to the other end of the sound path (from the triangle to the cross symbol) at a constant speed. However, you can accelerate or decelerate the movement of an ID, using Motion View.

### 7.6.1 Dome View

#### Creating a Sound Path

In Zirkonium, there are three ways to create a sound path. You can create it with a pen tool and draw a path manually using Bézier curves, let the software draw curves for you, employing a drawing algorithm, or, you can record the mouse motion in real-time in Dome View.

##### -Using Pen Tool

Select a single spatial event and click the pen tool at the top right corner of the window. Then, click in the dome view. The first click creates the start position of the sound path. The second click creates a tentative end position of the sound path. These two positions are marked with “triangle” and “cross” symbols respectively (fig:7.28).

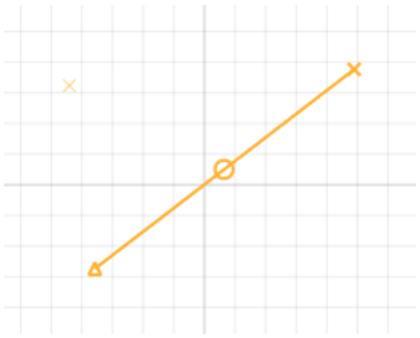

Figure 7.28: Triangle and Cross symbols

If you add more points to the path by clicking the dome view, the path will be extended and the end position of path moves to the newest point added.

To bend a sound path, you can drag the mouse cursor right after adding a point to the dome view. Alternatively, you can select the arrow tool and drag the Bézier handles.

##### -Using “Create circle / spiral” popover

It’s not easy to draw a perfect circle or spiral with Bézier curves, but Zirkonium offers a utility function that facilitates the creation of a circle- and spiral-shaped sound path.

To activate this function, press right mouse button or Ctrl + left mouse button in the dome view, then “add circle/spiral” context menu should appear. Simply click the “add circle/sprial” option to open the **add circle / spiral popover**.

In the popover, input the following four parameters that determine the property of a circle / spiral and press add button.

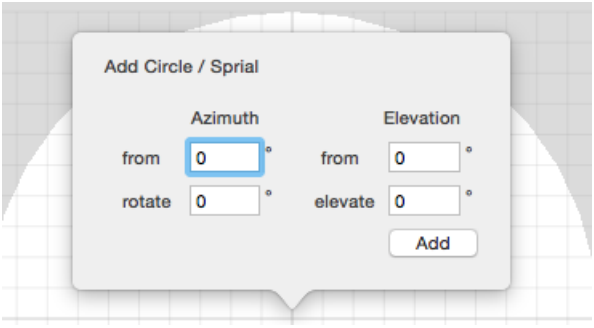


Figure 7.29: add circle/spiral popover

| | |
|---|---|
| Azimuth from | the origin point of the azimuth |
| Azimuth rotate | the amount of azimuth rotation in this event. |
| Elevation from | the origin of the elevation. 90 means that the ID is located at the zenith. |
| Elevation | The amount of elevation in this event. |

Table 7.6: Parameters of Add Circle / Spiral Popover

### Editing a Sound Path

#### -Moving Anchor Points

To move an anchor point, select **arrow tool** and drag the anchor point you want to move.

> You can snap your anchor point to the speakers or the grid by enabling “snap to speakers...” or “snap to grid...” options under View -> Dome View menu.

#### -Bending a Sound Path

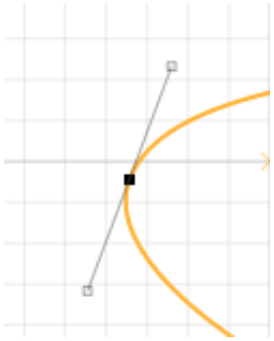

Figure 7.30: Bezier Handles

To bend existing Bézier curves, select arrow tool and drag the Bézier handles (fig:7.30), that stem from an anchor point. Two handles belong to one anchor point and each of them control different sides of the curve. By default, when you drag one of them, these two handles move together, but you can move a single handle by pressing shift and dragging. To reset a curve, click the handle with the option key pressed.

#### -Applying Affine Transformation to a Sound Path

With the affine tool, you can translate, rotate, or scale the entire sound path drawn in the dome view. In order to apply these operations, select the affine tool beneath the pen tool. Then, a bounding box with 9 small black handles, called **affine handles**, appears around the path.

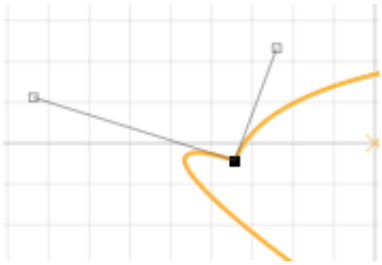

Figure 7.31: Bending one side by shift + drag

To translate the path, drag the handle in the middle. To rotate the path, drag the line of the bounding box. To scale the path, drag the 8 black affine handles surrounding the path.

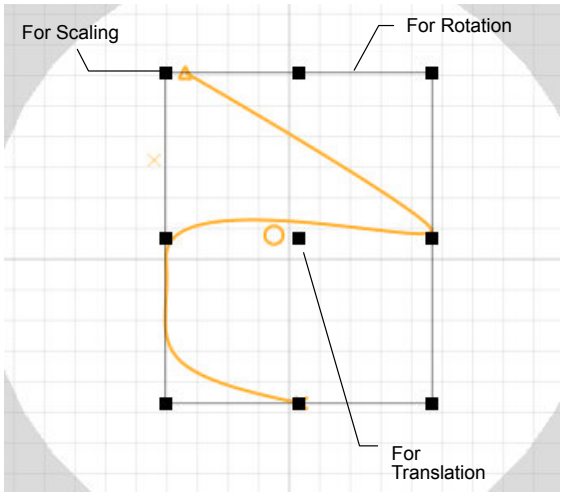


Figure 7.32: Affine Handles and their functions

#### -Snapping start or end position of a Sound Path

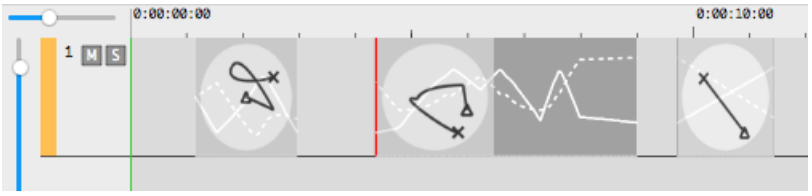


Figure 7.33: Incontinuity of sound paths

While playback, an ID instantly jumps to the start position of the sound path without any interpolation, when the playback cursor reaches the start time of a certain event, unless the end position of the previous event perfectly matches the start position of the current event, this will cause an abrupt change of ID position or “jump” at the beginning of an event that may cause unnatural auditive incontinuity, especially if an audible signal is played by the spatial rendering algorithm at the moment of the “jump”. Zirkonium automatically finds these “jumps” and indicates them with red lines (called Jump Warning Line) at the left edge of event regions (fig7.33).

There are four solutions to avoid this problem.

1. moving the start position of the current event to the end position of the previous event
2. moving the end position of the previous event to the start position of the current event

3. creating one more event between the current and previous event for interpolating the start position of the current event and the end position of the previous event
4. starting the event at the moment when the source (sound file or live input) produces absolutely no sound

Zirkonium is capable of performing above-mentioned solutions 1 - 3 automatically. To move the start position of the current event to the end position of the previous event, press the “snap to cross” button (fig:7.34) above the Camera Position Reset Button.

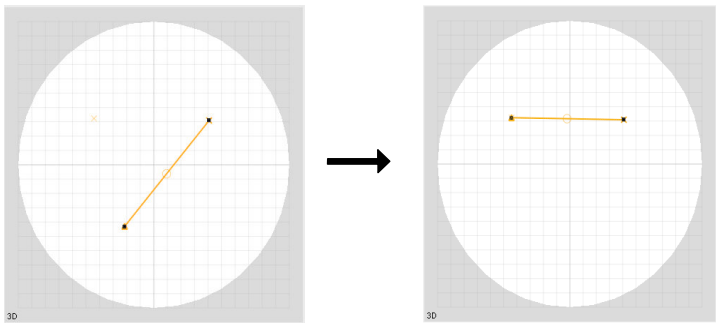

Figure 7.34: Snap To Cross

To move the end position of the previous event to the start position of the current event. Select the previous event in the event table or event view and press the “snap to triangle” button (fig:7.35).

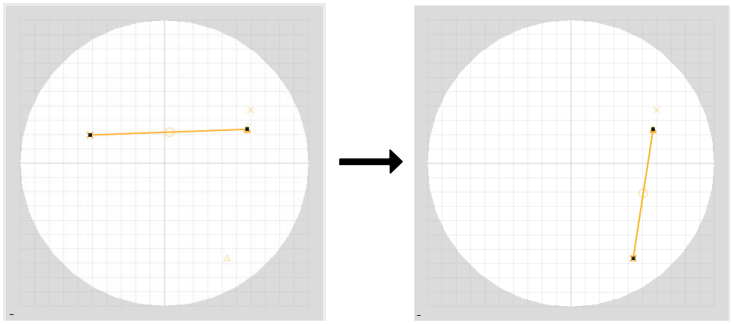

Figure 7.35: Snap To Triangle

To create an interpolation event, simply create an event between the current and the previous event with the pen tool. The newly created event will be automatically filled with a sound path, that interpolates the start position of the current event and the end position of the previous event (fig:7.36).

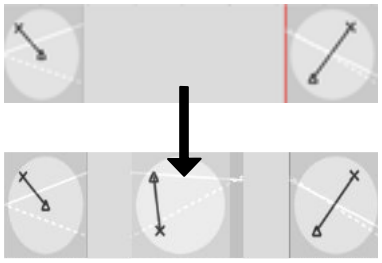

Figure 7.36: Interpolation Event

#### -Moving Camera Position

You can move the camera position vertically by rotating your mouse wheel and horizontally by shift key + mouse wheel. Zooming in/out is also possible by command key + mouse wheel. To reset the camera position, click the Camera Position Reset Button above the sliders.

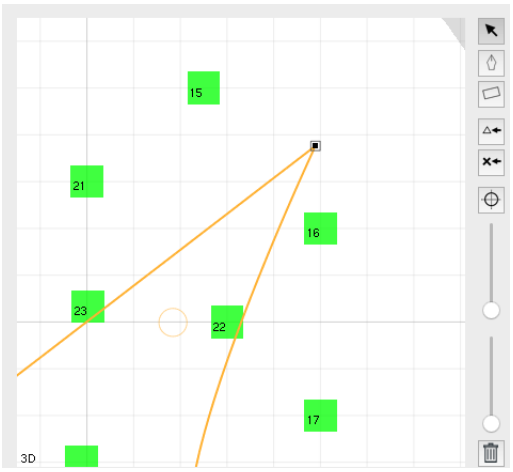


Figure 7.37: Zoomed Dome View with shifted camera position

#### -Deleting a part of a sound Path

To delete specific parts of a sound path:

1. Click on the dome view and drag the mouse. A selection rectangle should appear on the view.
2. When the mouse button is released, all anchor points within the selection rectangle turn red, meaning these anchor points are selected.
3. Press backspace to delete the selected anchor points. Click on an empty space in the dome view to deselect all selected anchor points.

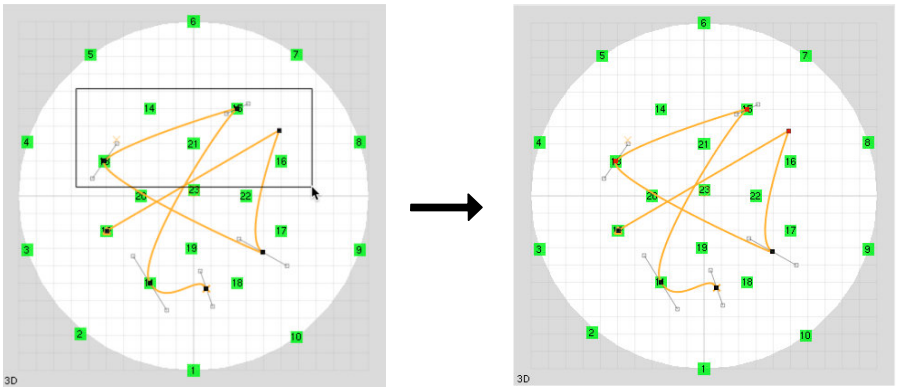

Figure 7.38: Selecting certain anchor points

#### -Deleting entire sound path

To delete a sound path in the dome view, simply click on the “trash” button located near the right bottom corner of the Dome View.

### 7.6.2 Motion View

You can control the movement of an ID along the defined sound path and expansion of sound diffusion with the Motion View. A motion path and a span path are automatically created, when an

event is created. The motion path becomes editable when more than one anchor points are added to the dome view. In other words, motion path is not available for an instantaneous event with a single anchor point or an undefined event with no anchor points.

#### Switching motion mode and span mode

By clicking **motion / span switch** located at the right of the motion view, you can change the current mode of the view (fig7.39).

Figure 7.39: In Motion Mode (Left) / In Span Mode (Right)

#### Editing motion path

Motion path defines the position of an ID along the sound path during the time frame of an event. By default, the ID moves from the start position (triangle symbol) to the end position (cross symbol) in a constant speed. However, you can bend the path and add extra anchor points to the path for realizing more complex movement of ID between the start and end position.

#### -Bending curves

To bend the motion path, hover the mouse cursor on the motion path with the arrow tool and the cursor will become an open-hand. Then, simply click the curve and drag it up or down to bend it.

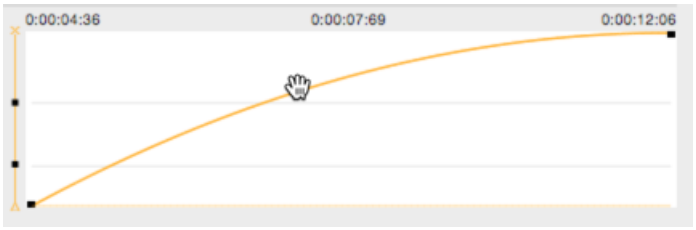


Figure 7.40: Bending Motion Path

#### -Adding anchor points

The position of the two anchor points provided by default can not be moved, since an ID has to move from the start position (triangle) to the end position (cross). However, you can add extra anchor points in the motion view by clicking the view with the pen tool (fig:7.41).

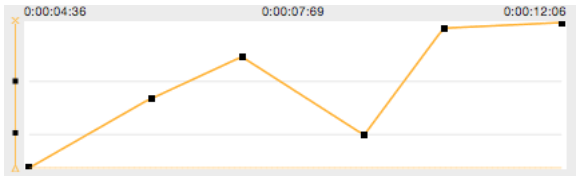


Figure 7.41: A Motion Path with multiple anchor points

#### -Adjusting the position of audio content to be played

By combining the Dome View and Motion View, you can adjust the position of audio content without changing the shape of the sound path. Here is an example:

In figure 7.42-left, a one-second sound material is spatialized. In this setting, the loudest part of this audio material will be mainly played by the speaker No. 20, if you do not edit the motion view

at all.

In case you want the loudest part of this audio material to be played from speaker No. 21, add one more anchor point and place the anchor point in the middle of the sound path on top of speaker No. 21. The added anchor point is also displayed in the Mini-Sound Path at the left side of the motion view (fig:7.42-center).

Then, add an anchor point in the motion view and place it at the loudest moment in the waveform and anchor point at the mini-sound path. This means that ID reaches the anchor point on the Mini-Sound path (i.e. speaker No. 21) at the loudest moment (fig:7.42-right).

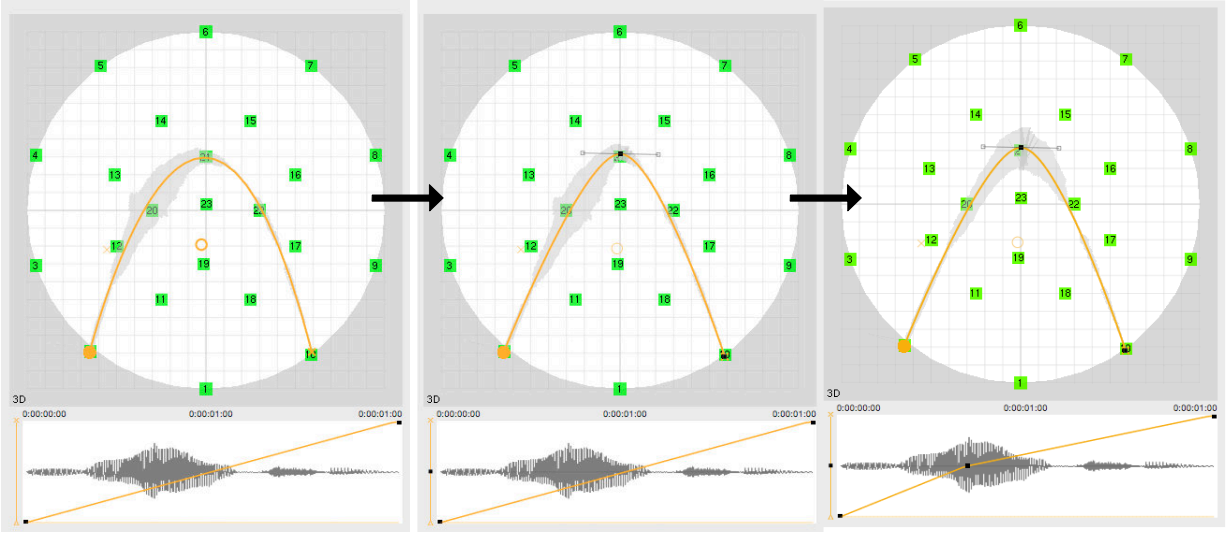

Figure 7.42: Adjusting audio content and position

In this way, you can manipulate flexibly the relationship between the position in Dome View and the audio content.

#### -Deleting a part of a motion/span path

To delete the added anchor points in motion view, select the arrow tool and draw a selection rectangle in the motion view. The selected anchor points in the selection rectangle turn red and can be deleted if the backspace is pressed (fig7.43).


Figure 7.43: Region Selection in Motion View

You can not delete the first and last anchor points.

#### Editing span path

The span path can be drawn in a similar manner. However, unlike a motion path, a span path doesn't have to reach the top-right corner of the view at the end time of an event.

#### Resetting the entire sound / span path

To reset the motion path and the span path at once, simply click on the “trash” button located at the right bottom corner of the Motion View.

# 8. Exporting Projects

In order to improve compatibility and to realize long-term storage, Zirkonium offers various exporting functionalities, SpatDIF, Bounce and Archive. This chapter describes each function.

## 8.1 Exporting SpatDIF

### 8.1.1 What is SpatDIF

According to the official website (`http://spatdif.org`), the definition of SpatDIF is as follows:

> SpatDIF, the Spatial Sound Description Interchange Format, is an ongoing collaborative effort offering a semantic and syntactic specification for storing and transmitting spatial audio scene description. The SpatDIF core is a lightweight minimal solution providing the most essential set of descriptors for spatial sound scenes. Additional descriptors are introduced as extensions, expanding the namespace and scope with respect to authoring, scene description, rendering, and reproduction of spatial sound.

Zirkonium has the capability of exporting trajectory events using the syntax of SpatDIF. For details of the syntax, refer to the website of SpatDIF.

### 8.1.2 Export as SpatDIF File

To export a SpatDIF-conformed XML file, simply select **File -> Export to SpatDIF** from the menu.

Zirkonium *samples* spatial events every 25 msec and exports these sampled trajectory data, using SpatDIF core descriptors. You can load the exported SpatDIF XML files onto Max or Pure Data patches, using SpatDIF external objects, available from ICST Zürich (`https://www.zhdk.ch/index.php?id=icst_spatdifexternal`). You can find a simple demo of SpatDIF export with a Max patch in the demo folder of the package.

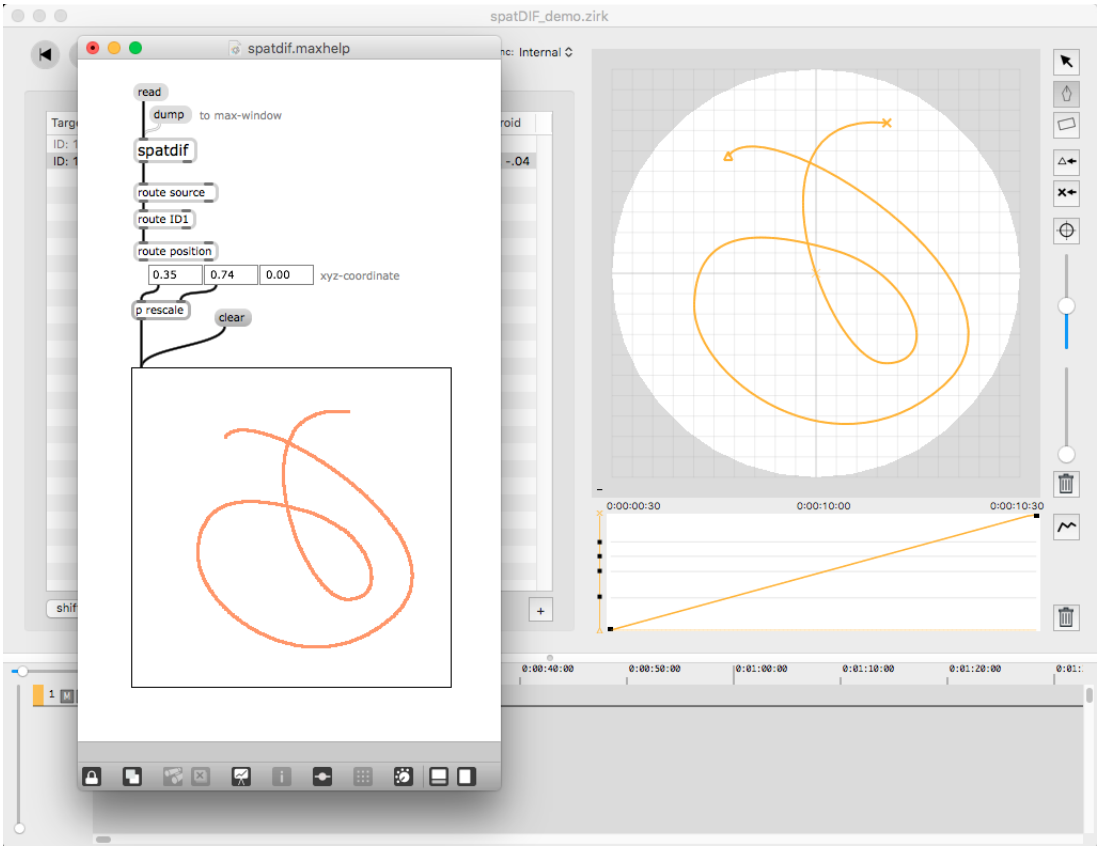


Figure 8.1: A trajectory designed by Zirkonium, loaded onto a Max Patch

## 8.2 Bouncing

Bounce function allows you to record all audio signals sent to loudspeakers and store them in sound files.

In order to activate the bounce function, select File->Bounce. Then, a save panel will appear. Select the destination folder, name the bounced file, and click the "Save" button. The modal sheet "bouncing in progress" will then appear on the window (Fig. 8.2).

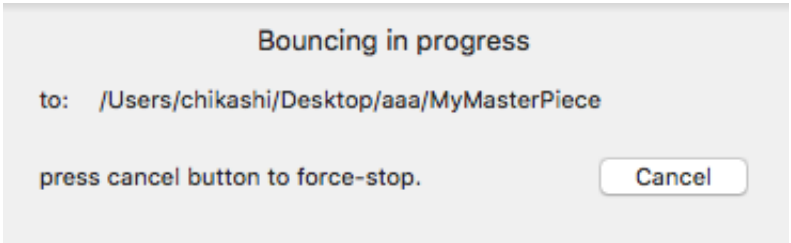


Figure 8.2: Bouncing in progress modal sheet

Note, the bounce function **does not** create a new folder automatically for the bounced files and the files are automatically suffixed by the speaker index number(Fig. 8.3).

The bounce function is currently "online", meaning, bouncing requires the duration of the piece. To interrupt the bounce, press the "cancel" button.

R The format of the bounced sound files is WAV 24 bit.

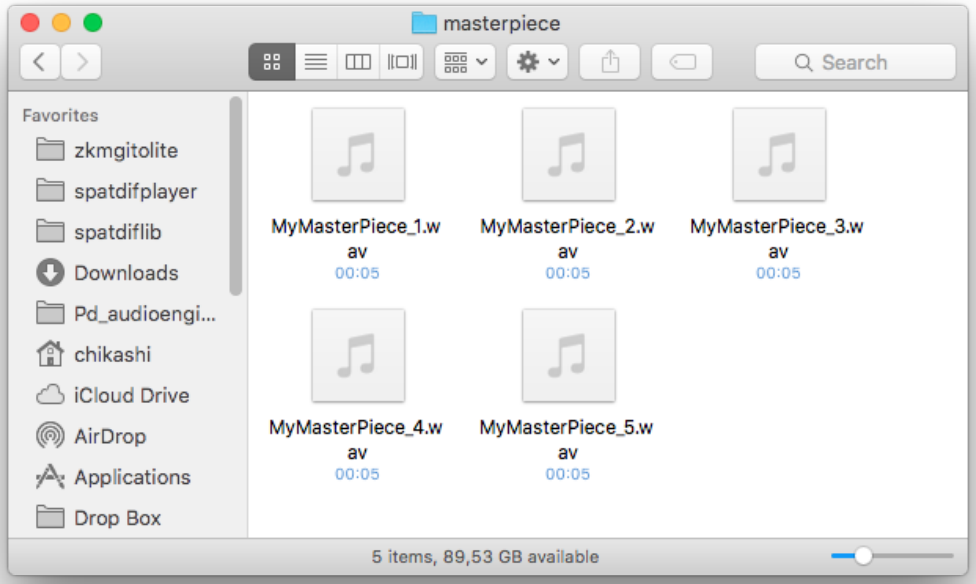


Figure 8.3: Bounced files are automatically suffixed

## 8.3 Archiving

You can import multiple sound files located at different folders on your disk system. Zirkonium stores their absolute paths and reloads this data again when the Zirkonium document is reopened. However, in case you want to copy your piece to another computer or send it to someone else, you may need to gather all your sound files scattered in your disk system and make a package that contains all data required to reconstruct your piece in another environment (This process is often called "consolidation"). Zirkonium offers a function to automatically execute this process. To do this, save your current project at least once and select **File->Archive Project**.

Zirkonium will then create a new folder under the folder where the Zirkonium file is located and name it *project name*_archive. Then, all sound files used in the project, as well as the Zirkonium file itself, are stored in this folder. The zirkonium file is automatically suffixed with "_archive."

You can compress the created folder and copy it to another machine or disk system.

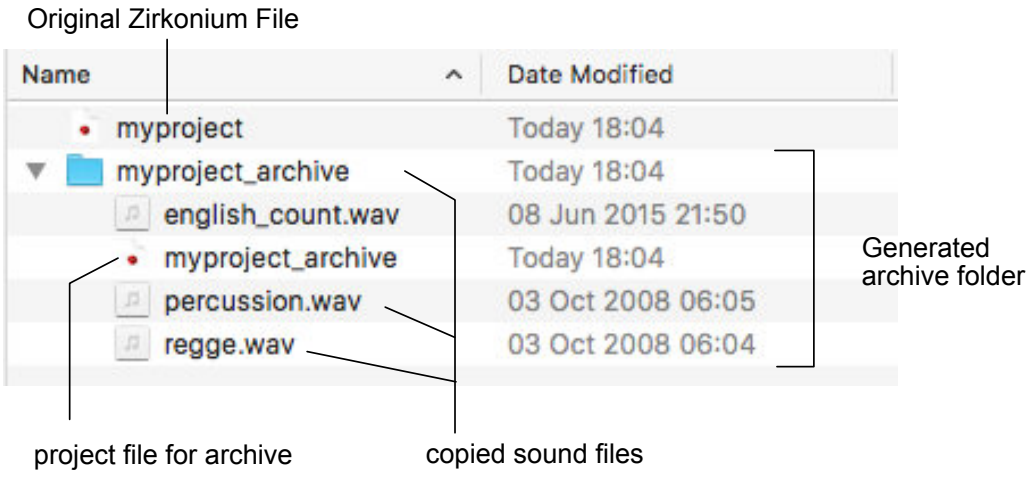


Figure 8.4: An example of archived project

# 9. Other functionalities

This chapter briefly introduces further functionalities of Zirkonium Trajectory Editor.

## 9.1 Importing MK1

To import a Zirkonium file created with "classic Zirkonium", select **File->Import MK1 File**. This imports all IDs and group properties as well as their events.

## 9.2 Controlling with OSC

The position and span of each ID can be controlled by other applications, such as Max, Pd or SuperCollider via OSC message. All configuration in regard with OSC message traffic can be configured in the “Network Settings sheet”. This sheet is accessible from File > Network Settings menu 9.1.

The left side of the sheet configures the internal OSC senders in the Trajectory editor that send spepcific data in real-time to external software or hardware. The right side of the sheet configures the internal OSC receiver that receives OSC messages from an external software. You can have multiple senders within the Trajectory editor but only one receiver.

### 9.2.1 OSC Sender

The left side of the sheet configures the internal OSC senders.

#### OSC Sender table

The table shows the currently active OSC senders. You can configure the following parameters for each sender:

**IP address** The IP address of the device that messages are sent to
**port** The port of the device that the sender send messages to
**Time** If checked, the time message will be sent
**IDPos** If checked, the position of each ID will be sent

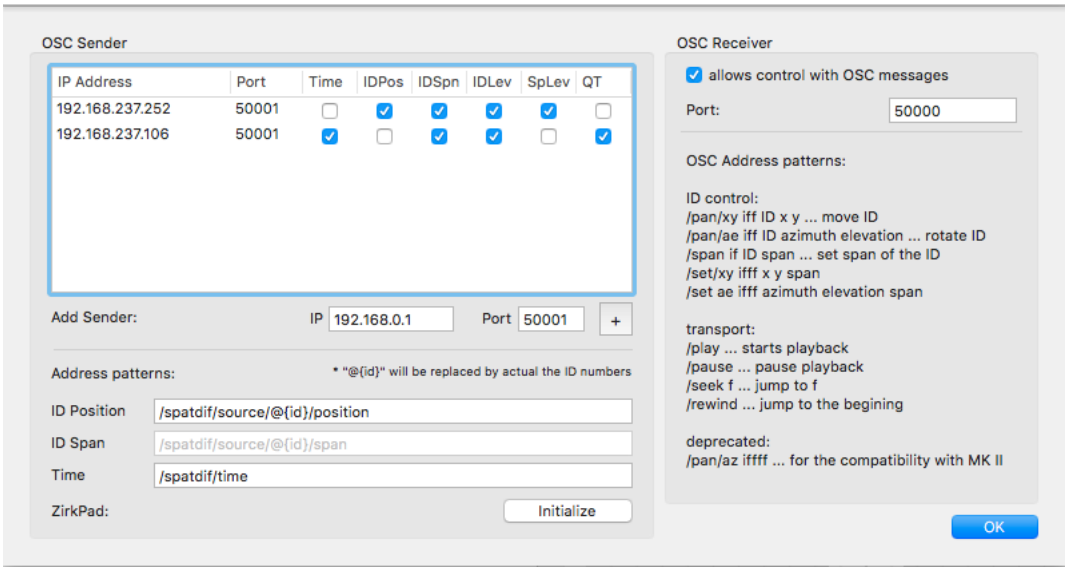


Figure 9.1: Network Setting Sheet

**IDSpn** If checked, the span of each ID will be sent
**IDLev** If checked, the current level of each ID will be sent
**QT** If checked, Zirkonium sends messages meant for ZirkQTplayer
IP address and port fields become editable by double-clicking.

#### Add Sender fields

You can add new senders by entering an IP and Port number in the "Add Sender" fields and clicking the "+" button.

#### Address Patterns

These four fields determine the OSC address pattern of the ID position, ID span and time messages. @{id} in the address pattern will be replaced with the ID number when a message is sent.

#### Initialize button

One this button is clicked, the Trajectory editor sends the ZirkPad initialization message, containing definitions of all IDs and groups to all devices listed in the OSC sender table.

### 9.2.2 OSC Receiver

The right side of the sheet configures the internal OSC receiver.
**"allow control with OSC messages" check box** when this check box is checked, the Trajectory editor accepts incoming OSC messages
**Port** The OSC port used for receiving OSC messages
**OSC Address Patterns** the list of OSC messages that the Trajectory editor currently accepts

## 9.3 ZirkOSC3

ZirkOSC3 (Fig. 9.2) is an AU/VST plugin developed by Robert Normandaue for controlling Zirkonium remotely from a DAW software via OSC protocol.

### 9.3.1 Download and Installation

ZirkOSC3 is available under `https://sourceforge.net/projects/zirkosc3/`. Uncompress the package and place the VST and AU-component under /Library/Audio/Plug-Ins/VST and

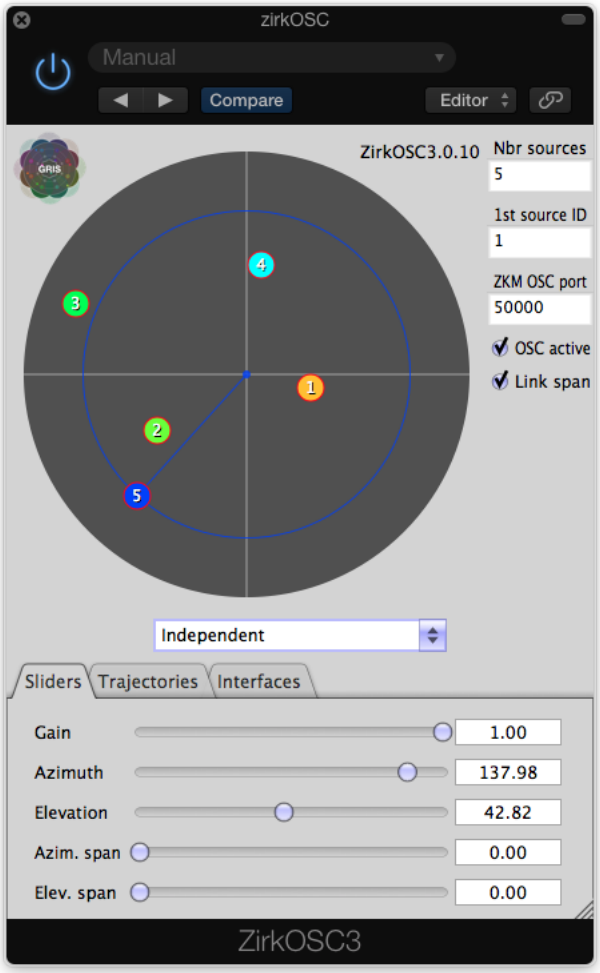


Figure 9.2: ZirkOSC running on Logic Pro

Components, respectively.

### 9.3.2 Network Setting

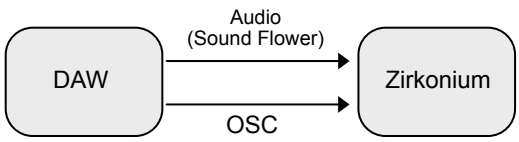


Figure 9.3: DAW-based spatialization system, using Zirkonium as the renderer

In order to receive OSC messages from ZirkOSC, the OSC receiver must be activated. Open the Network Setting ( **File -> Network Settings**) and check if the OSC receiver is activated.

R Most composers combine ZirkOSC and "audio piping" software such as SoundFlower, in order to send not only OSC messages but also audio streams from the DAW. In this way, the user is able to utilize Zrikonium Trajectory editor solely as a spatialization engine or a "renderer" for the DAW software (Fig. **??**).

## 9.4 Syncing with other software

Zirkonium is able to synchronize with other Audio software, such as Logic, Ableton Live or Max, using MIDI Time Code or OSC. You can select a mode for synchoronization with a pop-up button beneath the Transport bar.


Figure 9.4: Sync mode pop-up button beneath the Transport Bar

The followings are the descriptions of each mode.

#### internal

Trajectory Editor doesn't send or receive any messages for sync.

#### MTC Slave

Accepts MTC message from other software and adjust current time to the received messages. If this mode is selected, play button will be immediately disabled and become unclickable.

> The MTC Frame Rate that Zirkonium sends in MTC Master mode is fixed at 30 fps and currently, is not modifiable.

> In MTC Master Mode, Zirkonium sends **Full Frame MTC Messages** when the user manually inputs a new time to the current time field or clicks on the time view. Zirkonium sends **Quarter Frame MTC Messages** during playback.

»»»> 044d455ba68f450b238e5d2a7d78f68046e2d50c

## 9.5 Customizing Spatialization Server

R This section is intended for advanced Pd users. If you are not familiar with Pd and would like to modify/customize the spatialization algorithms of Zirkonium. Please refer to the Pd website `http://puredata.info`.

### 9.5.1 How Zirkonium realizes spatial rendering

Zirkonium internally utilizes Pure Data as the audio processing engine, and this core functionality is integrated in the Zirkonium Trajectory Editor, employing libPd.

You are able to access the **Zirkonium Spatialization Server**, a Pd patch that processes the sound files and executes the spatialization algorithms, and customize it AT YOUR OWN RISK.

### 9.5.2 Accessing Zirkonium Server Patch

In order to access the Zirkonium Server Patch, right click on the ZirkoniumTrajectoryServer application icon and choose "show package content"[Fig:9.5]. You will find the core Pd patch named "zirkonium_server.pd" as well as a few abstractions and external objects under Contents/Resources folder.

nc

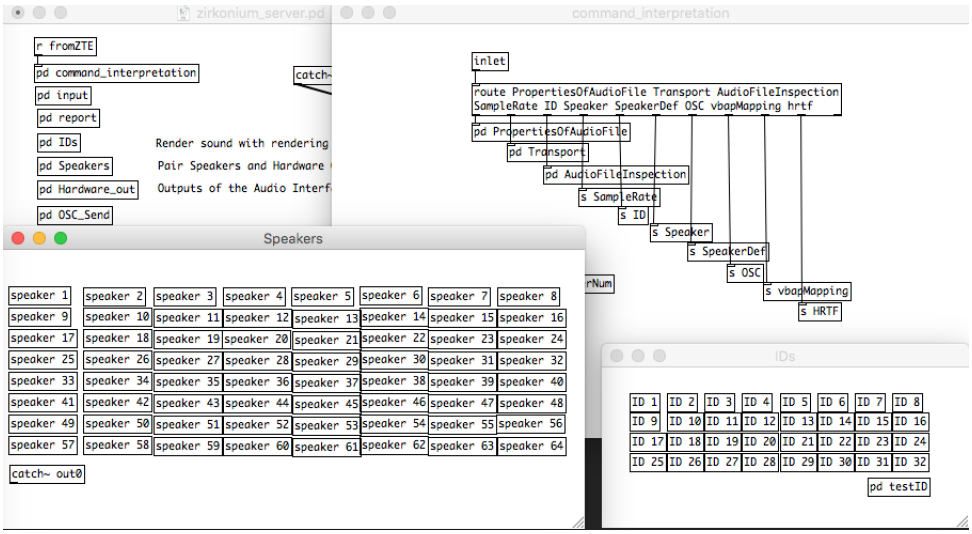


Figure 9.5: Location of Zirkonium Server

# 10. Appendix 1:ZirkPad

## 10.1 What is ZirkPad?

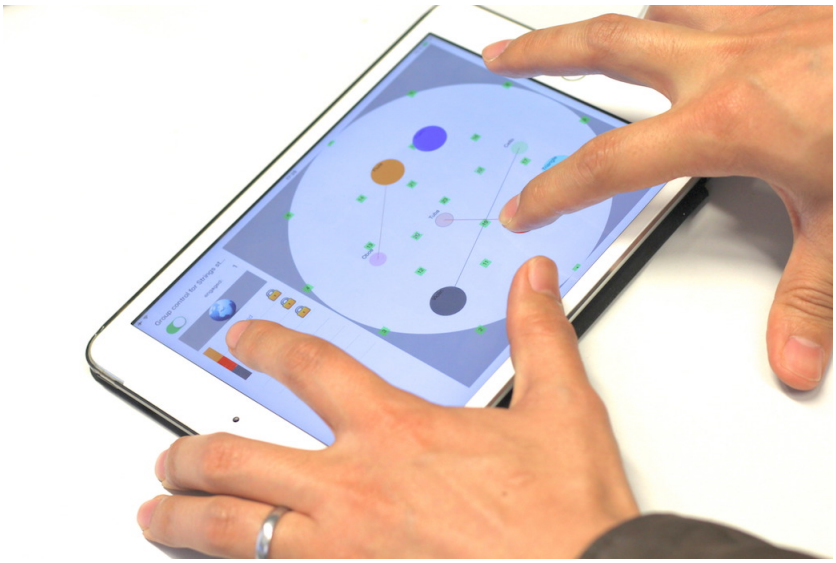

Figure 10.1: Live spatialization with ZirkPad

ZirkPad is an iPad App that enables users to control the position and the span of each ID or group of the Zirkonium Trajectory editor with the multi-touch interface remotely. This app is developed mainly for three scenarios.

**Live Diffusion of more than two channels**

With ZirkPad, you can spatialize up to 64 channels for sound sources and control the movement of each sound source directly by dragging the virual sound object displayed on the multi-touch-screen. In this way, ZirkPad offers an alternative live-spatialization strategy in contrast to the traditional "diffusion", a live expansion of stereo fixed-media using a mixing console and multiple stereo-pairs

of speakers arranged in a listening space.

#### Sound reinforcement for a surround speaker system

Zirkonium trjaectory editor is capable of patching live inputs and IDs, thus, ZirkPad can be used for live performance with musicians. In this way, ZirkPad can assist the PA-engineer to "place" the audio signals coming from each instrument onto a surround loudspeaker system.

#### Listening and controlling at various positions

Sound impression varies significantly depending on the position of the listener and all concert guests can not listen to the sound from the sweetspot simultaneously. Thus, we are often required to listen and evaluate how the sound source is heard at the various positions of the listening space. Beacuse ZirkPad runs on an iPad and sends message wirelessly to the Trajectory editor, you are able to walk around the listening space and control the position of sound sources at the same time. This significantly reduces the time for audio configuration before a concert.

## 10.2 System requirement

ZirkPad requires the following hardware/software environment.

### 10.2.1 Hardware Requirement

All iPads except the first generation iPad.

### 10.2.2 OS

iOS 9 or higher.

## 10.3 Network preparation

An iPad or multiple iPads, running ZirkPad, must be connected to the same local area network (LAN) of the host computer, runing Trajectory editor as shown in figure 10.2.

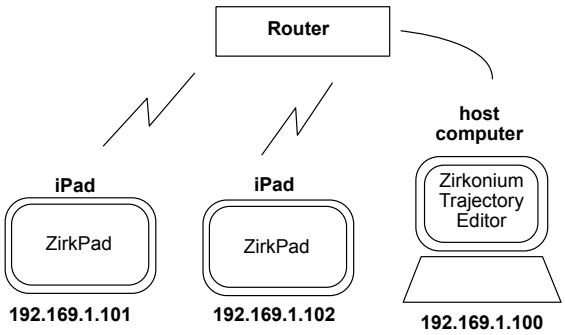


Figure 10.2: ZirkPad Network

R It is highly encouraged to use a dedicated router, assign static IP address for each device, and use non-crowded WIFI channels for live performances.

## 10.4 Installation

You can download and install the application from the App store.

## 10.5 Initialization

The following information will be sent from the Trajectory editor to the ZirkPad. It is highly encouraged to name all your IDs and groups before sending initialization message from the Trajectory editor to the ZirkPad.

- ID name
- ID color
- ID initial position
- Group name
- Group master
- Speaker positions

### 10.5.1 OSC sender setting

In order to send OSC messages from the Trajectory editor to ZirkPad, an OSC sender must be defined in the Trajectory editor. To create a new OSC sender in the Trajectory editor, open the Network Settings sheet by selecting File->Network Settings menu.

Figure 10.3: ZirkPad shows the IP address of iPad on startup

| IP Address | Port | Time | IDPos | IDSpn | IDLev | SpLev | QT |
|---|---|---|---|---|---|---|---|
| 192.168.237.252 | 50001 | ☐ | ☐ | ☐ | ☑ | ☑ | ☐ |

Figure 10.4: Typical Setup for ZirkPad

1. Make sure the OSC receiver on the right side of the sheet is activated and the port number is set to **50000**.
2. Launch the ZirkPad App on the iPad and check the IP address shown in the dome view on the right side (Fig. 10.3) .
3. Enter the IP address of the iPad in the “add sender” field at the bottom of the OSC sender table.
4. Set the port number to **50001** and press the “+” button
5. A new sender will be created in the table

6. Check the boxes **IDLev** and **SPLev**. By checking these check boxes the Trajectory editor sends the level of each ID and speaker to the ZirkPad. You can also deactivate these two in order to reduce the number of OSC messages.
7. Click the initialize button “init” at the bottom of the OSC Sender section.

When the initialization is successful, loudspeakers and IDs will appear in the Dome view. If not, you will receive an error message. If you receive an error message, please click the “initialize” button again 10.5.

If you change the configuration of IDs or groups in the Trajectory editor, you must send the initialization message again in order to update ZirkPad.

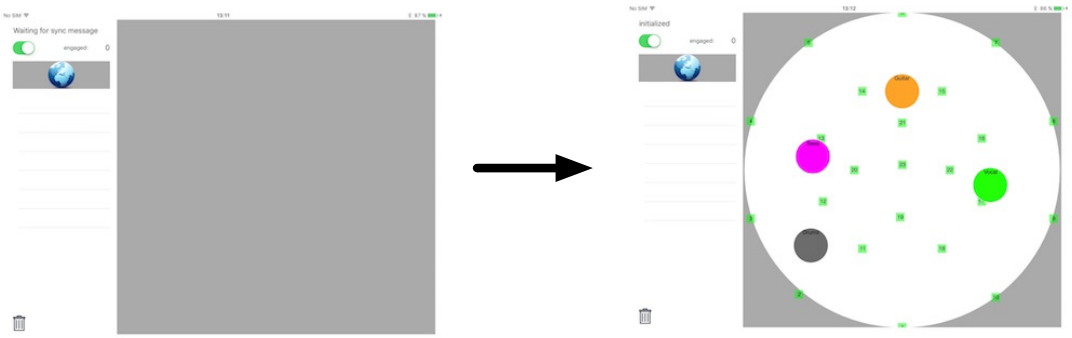

Figure 10.5: Initialization

If ZirkPad doesn’t react to the initialization from the Trajectory editor, check the network settings of both devices.

## 10.6 GUI Overview

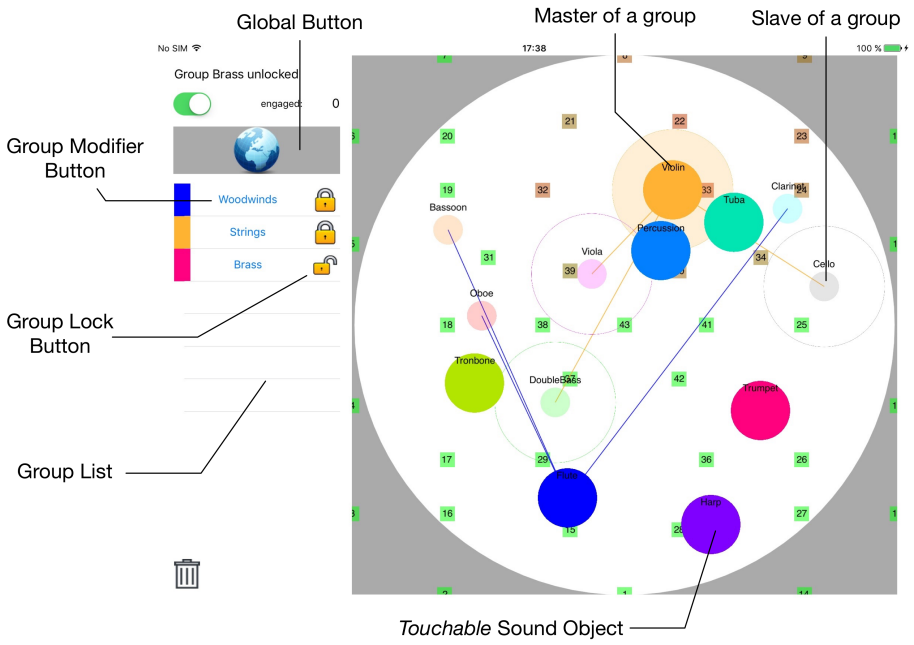


Figure 10.6: Zirk Pad GUI

The figure 10.6 shows the main GUI of the ZirkPad. On the right side of the screen, the multi-touch-enabled Dome view is displayed. It synchronizes with the Dome view of the Trajectory editor and displays the position of speakers and sound objects as well as the levels of the audio signals that each sound object generates and each speaker receives.

All IDs are labeled with the their names. The group networks are displayed with thin straight lines. By default, all slaves are visualized with small circles and they are not controllable.

The left side of the window shows several buttons and switches to activate group-based functionalities.

#### Global Lock Switch

This switch activates/deactivates all functionalities that the left pane of the ZirkPad offers. By deactivating the functions in the left pane, you can avoid unintended manipulations caused by touching the left pane.

#### Global Button

If this button is pressed, all IDs in the dome view will be controlled parametrically (see 10.8.2 for details)

#### Group Button List

The group list shows the names of ID groups defined in groups table in the Trajectory editor. The colors of the small rectangles next to the group names are the color of the Master ID of each group.

#### Group Modifier Button

When a name of group is pressed, ZirkPad enters "Group mode" and allows you to control the designated group in various ways. For details, refer to section **??**.

#### Group Lock Button

When the group lock button is activated (locked), you can not control the position of slaves in a group. But once it is unlocked, you can control the position of each ID individually and change the formation of the group.

## 10.7 Single ID Manipulation

Unlike the Dome view in the Trajectory editor, the sound objects in the Dome view in ZirkPad are *touchable*; you can move ID objects that do not belong to any groups by dragging them with your fingers. As soon as ZirkPad recognizes touches on sound objects, it sends OSC messages to the Trajectory editor. In response to the received OSC messages, the Trajectory editor updates the position of the sound objects immediately.

In order to control the span (diffusion) of an ID, first touch the ID circle with one finger, then place another finger on a empty part of the dome view, and drag it along the Y-axis. Then, a thin circle will appear around the target ID. This circle indicates the current span of the ID.

## 10.8 Group-based Manipulation

If you want to control multiple IDs at once, ZirkPad offers two options:*Direct mode* and *Parametric mode*.

### 10.8.1 Direct mode

Your can control the position of each ID just by directly dragging the master ID in the dome view. Slave IDs follow the master ID using the rotate mode described in chapter 6. Note, slaves of a group cannot be touched by default because all groups are "locked". However, you can unlock the

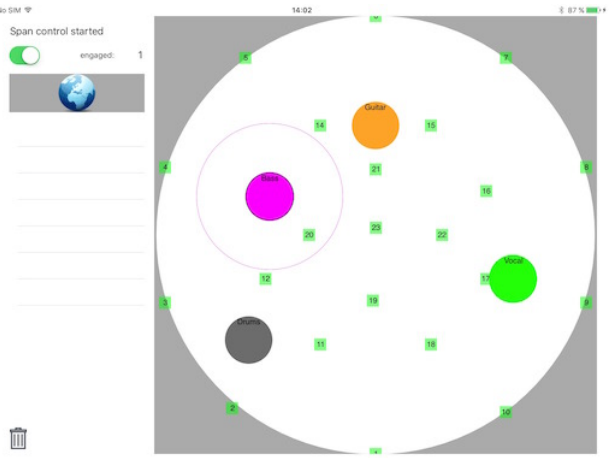


Figure 10.7: The Span circle indicated around an ID

group by tapping the lock button next to the group name. If you move the position of the slaves and lock the group again, the formation of the group will be changed.

### 10.8.2 Parametric mode

By touching the name of a group(s) in the group list, you enter in parametric mode. You can exit this mode by simply releasing your finger from the group list.

In the parametric mode, your finger gestures in the dome view control the position of IDs that belong to the pressed group(s) *relatively*. The movement of the group is determined by the number of fingers you use in the dome view.

#### One Finger: Diffusion

If you use one finger and drag it in the dome view along Y-axis, you change the span of all designated IDs.

#### Two Fingers: Distance

If you use two fingers and swipe the dome view along the Y-axis, you control the distance from the center of all member IDs.

#### Three fingers: Rotation

If you use three fingers and swipe the dome view, you rotate all the selected IDs.

#### Four fingers: Translation

If you use four finger and swipe the dome view, you move all the selected IDs by translation.

By touching the global button above the group list, you can control all existing IDs and groups using the parametric mode.

## 10.9 Multiple manipulators

As the figure 10.2 shows, two or more manipulators can control one Trajectory Editor simultaneously.

However, if multiple manipulators are allowed to control all IDs, they may send different commands for the same ID at the same time, and it may result in abrupt “jump” of the ID position.

In order to avoid this, you can limit the number of groups that one manipulator can access. After the initialization, each manipulator can remove specific groups from the group list by tapping

groups and the trash button at the same time. In this way, you can isolate the control of multiple manipulators completely and avoid conflicts.

# 11. Appendix 2: Speaker Setup

Speaker Setup is a Mac OSX Software, that enables you to set up your own non-standardized loudspeaker configurations. The application lets you graphically organize the speakers and export the configuration as a XML file. The exported XML can be loaded onto the dome view of Zirkonium.

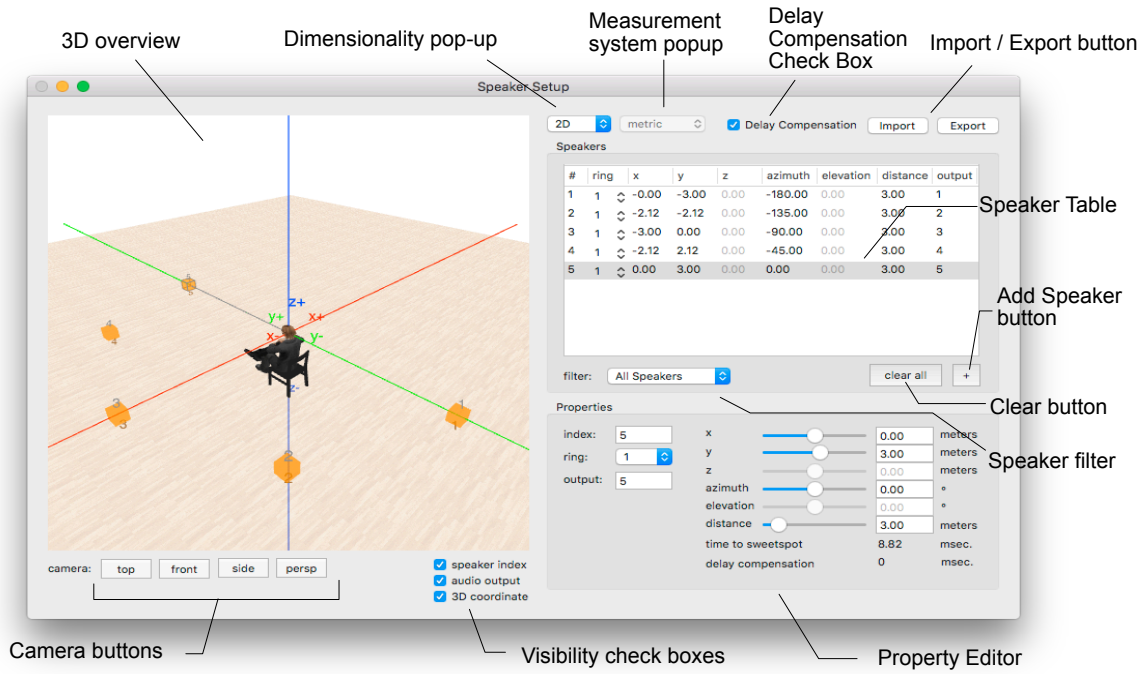


Figure 11.1: Speaker Setup

## 11.1 Installation, System Requirements

For the installation, just copy the Speaker Setup to your Applications folder. The application runs on Mac OS 10.9 or later.

## 11.2 GUI Overview

### 11.2.1 3D overview

This view shows a listener, surrounded by speakers, in a 3D space. You can change the camera position by dragging the view and zooming in/out by rotating the mouse wheel.

In the view, each speaker is indicated as a colored cube. The speaker that is furthest from the sweet-spot is indicated with a darker color than the other speakers. This speaker is used as the reference speaker for calculating the delay compensation.

The black numbers above the speakers indicate the index number of each speaker, and the brown numbers below the speakers are the numbers of the associated audio hardware outputs.

### 11.2.2 Camera buttons

By these buttons, you can move the camera in the 3D overview to a specific position instantly.

### 11.2.3 Dimensionality pop-up

You can define the dimensionality of your speaker setup. 2D means all speakers are positioned at the same height (ear level). If you select 2D, the z and elevation values of all speakers are instantly set to 0 and become uneditable.

### 11.2.4 Measurement system pop-up

In the speaker setup, you can define the position of speakers either with metric values or relative values from -1 to 1. This pop-up allows you to select a mode from these two options. This pop-up button will be disabled when you add the first speaker to the view. To enable this pop-up button again, delete the all speakers in the view.

R In case you select the relative mode, Speaker Setup will not output XML attributes for delay compensation.

### 11.2.5 Delay Compensation Check Box

This check box enables or disables the calculation of delay compensation based on metric distances between the speakers and the sweetspot.

### 11.2.6 Import / Export button

These buttons let you export and import XML data that contains a loudspeaker configuration.

### 11.2.7 Speaker table

This table lists all the speakers in the 3D overview. If you select a speaker in the list, a black line appears in the 3D overview to indicate the selected speaker.

You can delete the speaker by selecting a speaker and pressing the delete key.

The following is a description of each column of the table

**#**

Index number of each speaker. The number must be unique.

#### ring
The ring (i.e. speaker group) that the speaker belongs to. You can choose from ring 1 to 10.

#### x,y,z
The position of the speaker in Cartesian coordinates. When you change one of these values, the value of azimuth, elevation, and distance will be automatically recalculated.

#### azimuth, elevation, distance
The position of the speaker in spherical coordinates. When you change one of these values, the value of x, y, and z will be automatically recalculated.

#### Output
The audio hardware output channel assigned to the speaker.

### 11.2.8 Speaker filter
With this pop-up button, you can filter speakers shown in the table by the ring index. This function is particularly useful when you want to focus on a small group of speakers.

### 11.2.9 Clear All button
Clear all speakers in the scene.

### 11.2.10 "+" button
Add a new speaker to the scene.

> Adopting the convention of the first version of Zirkonium, Speaker Setup adds the first speaker right behind the listener and gives it the index No.1. Then, it labels adjacent speakers counter-clockwise by default. However, you are not obliged to follow this convention. If you prefer another numbering convention, feel free to change the speaker index number.

### 11.2.11 Speaker Property Editor
In this pane, all properties of the selected speaker are displayed and the values are adjustable with either sliders or text boxes.

If the relative measurement system is selected, and delay compensation is activated, two additional parameters are displayed at the bottom of this pane.

#### time to sweet-spot
Based on the provided data of speaker position, the software calculates automatically the approximate time it takes the sound to travel from the selected speaker to the sweet-spot. The speed of sound is predefined as 340 meters/sec. If the relative measurement system is selected, this field will be hidden.

#### delay for compensation
Delay compensation time based on the metrical position of each speaker. If the relative measurement system is selected or the delay compensation in unchecked, this field will be hidden.

> If the distance between the sweet-spot and all speakers is not same, the sound generated by nearer speakers from the sweet-spot reach the sweet-spot earlier than speakers that are further away. This may cause an unintended Haas effect and possibly alter the sound quality. In order to avoid this problem, Zirkonium is able to apply a slight delay to the signal fed to nearer speakers. This is called **delay compensation**. According to the provided metric distances of speakers,

> Speaker Setup automatically calculates the appropriate delay time for each speaker and stores them in the XML file.

R In the absolute mode, the range of x, y, z position sliders in the property editor is limited from -25 to 25 (meters). If you want to setup loudspeakers for a larger space, please type the number directly in the text box.

## 11.3 Typical Setup procedure

The following is an example of a typical speaker setup procedure:

1. Define the dimensionality of your speaker setup with the dimensionality pop-up. If the vertical position (i.e. the distance between the speaker cone and the floor ) of all your speakers in your room is equal, select 2D, if not, select 3D.
2. Select mode from absolute or relative with the measurement system pop-up. If the data of distances between sweet-spot and each speaker is at hand, select absolute mode. If the distance between the sweet spot and each speaker is not always equal, it is advised to choose absolute mode and measure the distance between the sweet-spot and the speakers.
3. Add an arbitrary number of speakers by clicking the “+” button several times.
4. If you want to categorize the added speakers into some groups, use “ring” pop-up menu in the speaker table and assign a new ring index to each speaker. You can filter the list by these ring indexes, using the speaker filter pop-up.
5. Select the created speakers one by one and enter the position of each speaker, using the sliders or the text field in the property editor.
6. When you finish with the setup, click the “export” button, name the XML file, and click the save button.

# 12. Appendix 3: ZirkoniumQTPlayer

ZirkoniumQTPlayer is a simple Quicktime player that accepts several simple OSC messages for playback. This enables you to synchronize a video and a soundtrack that is spatialized with Zrikonium.

## 12.1 Installation, System Requirements

For the installation, simply copy the Speaker Setup to your Applications folder. The application runs on Mac OS 10.5 or later.

## 12.2 Opening a Quick Time File

To open a quicktime file, select “open” from the file menu and select a quicktime movie in the open panel.

## 12.3 Preferences

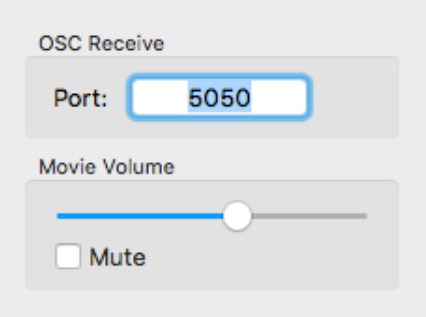


Figure 12.1: Preferences Panel

The preferences panel (fig:12.1) is accessible from **ZirkoniumQTPlayer -> Preferences...** menu. In the panel, you can specify a network port for receiving incoming OSC messages. The default port number is 1024 but you can change the port with the text field. The movie volume slider controls the gain of the movie's audio tracks. You can mute the audio instantly by checking the Mute check box.

## 12.4 Synchronizing with Zirkonium Trajectory Editor

To synchronize QTPlayer with the Trajectory Editor, the Trajectory Editor should send OSC messages to the QTPlayer. For the configuration open the network settings sheet by selecting **File -> Network Settings**.


Figure 12.2: Adding OSC Sender that sends messages to QT

In the "add sender" field in the Network Settings sheet. Enter the IP address of the computer on which the QTPlayer is running and the configured Port. If you run the QTPlayer on the same machine, enter 127.0.0.1 (loopback) in the IP address field and press the "+" button (Fig. 12.2).

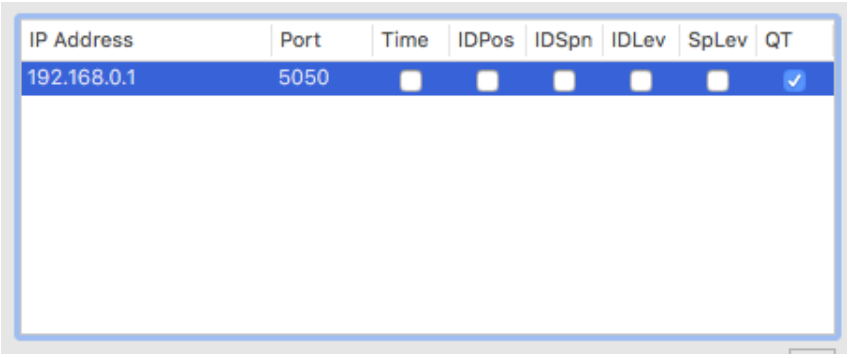


Figure 12.3: Check the QT column

Then the created OSC sender will be displayed in the OSC Sender table. In the table, check the QT column (Fig. 12.3). Now the Trajectory editor is able to send OSC messages to ZirkQTPlayer.

R ZirkQTplayer starts the playback simply when it receives a OSC message. There are no mechanisms to evaluate the synchronization or to correct it during the playback.

R It is also possible to send OSC messages to multiple computers in the same network and start QT movies on several computers simultaneously. To do so, define multiple OSC senders in the OSC Sender table.

# Index