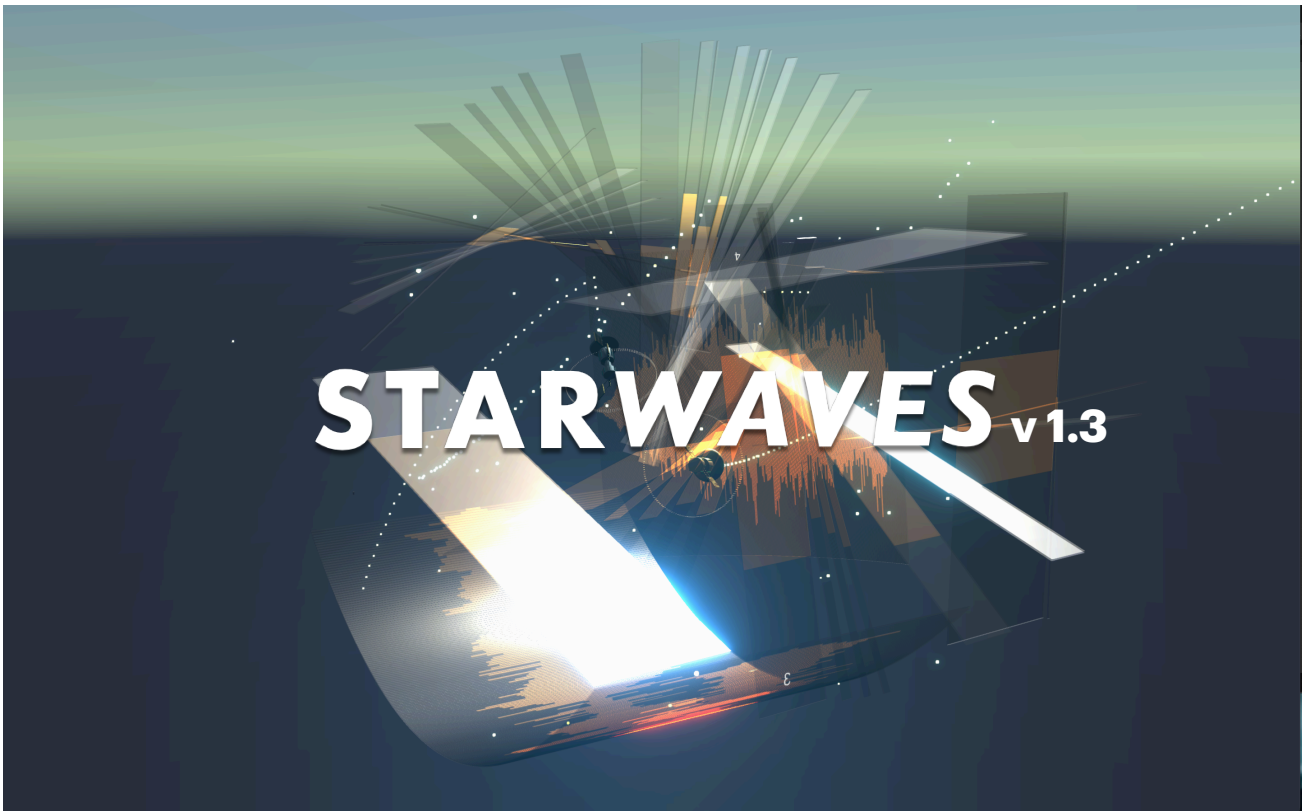


# SONIC PLANET

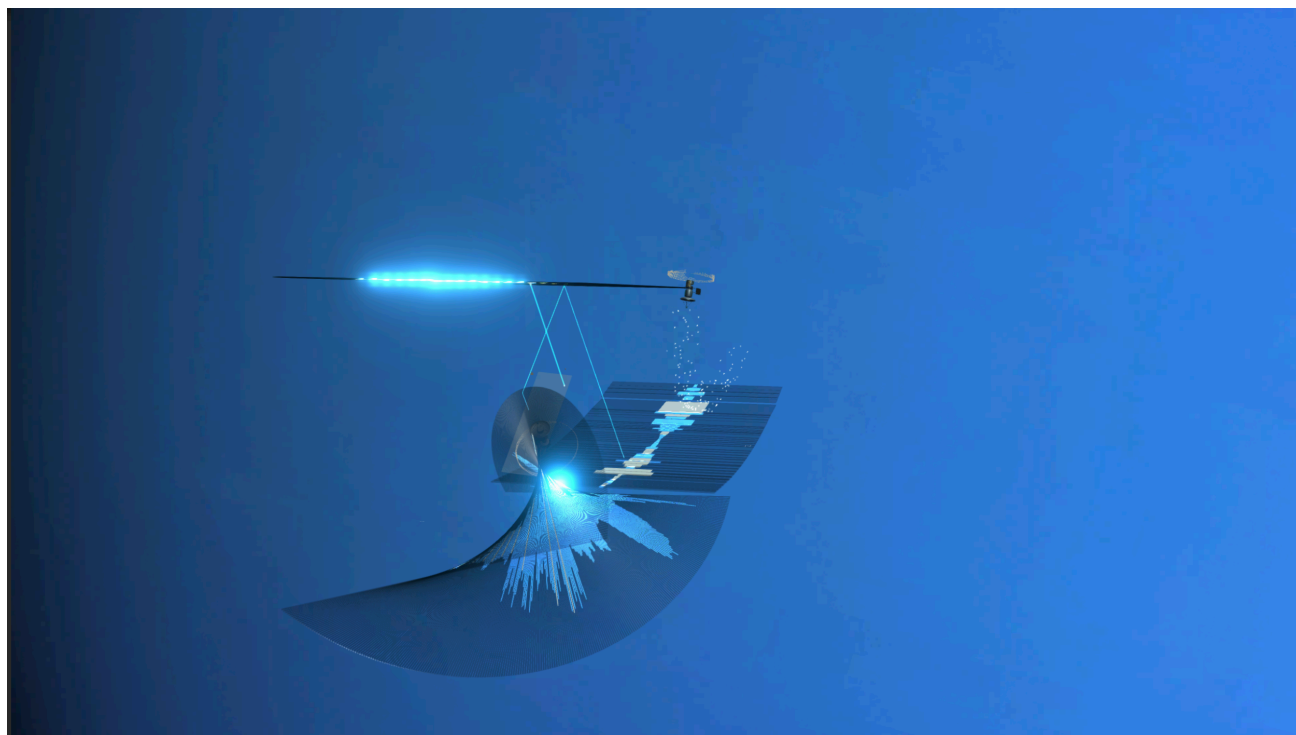


**STARWAVES** IS DESIGNED & PROGRAMMED BY **SINAN BÖKESÖY**.

**ALL RIGHTS RESERVED**

[WWW.SONICPLANET.COM](http://WWW.SONICPLANET.COM)

# WHAT IS STARWAVES



StarWaves is an audiovisual scene architect, the embodiment of sound, space and visual design rolled into one.

StarWaves gives the sound designer the architect's role by creating possibilities of dialogue between these multiple elements.

## IN THE MAKING OF SONIC ARCHITECTURES

StarWaves can import the sonic material in its concrete form as recorded files. Then this digital data is being represented as 3D rigid physical forms in space. We name them "the platform blocks" distributed spatially, a structure which stems from the sonic material being imported. There can be 100s of them, reaching the granular levels of sonic material. The layout of the platforms can be designed with 3D tools like positioning, rotation; they can be stretched, bent, scattered along their physical dimensions in continuous transformations.

As Goethe wisely stated : the architecture itself is "frozen music". How do we interact with this architecture to create the sonic textures ?

We know that sound travels as waves in an environment. It originates from a source and travels towards a distance, then interacts with the surface material of the environmental objects, gets

reflected, refracted, and absorbed within all the physical reality of its nature, granting a flexible relation to spatiality.

Sound remains as a hidden existence until it reaches our body. StarWaves uses on its scene the transmitters which send particles and beams of light to the space ( reminiscent to particle behavior of light or sound ) They hit the platform blocks ( representing the sonic material ) get reflected and collide with further objects, travel in distance until they die.

The time they collide with these rigid bodies they trigger sonic events and start a flow of synthesis processes. It is this continuous kinematic interaction, this physical realm which composes the sonic being and applies continuous transformations on its texture.

## **VISUAL, CAUSAL AND NARRATIVE FIDELITY OF MOTION**

The poetic expansion of architectural elements comes with the contribution of mathematics, geometry, physical laws ( kinematics ) altogether what defines this environment. The rich motion dynamics constitute a choreographed occupation of the StarWaves space, filled with actions triggering sonic events and with a formal beauty of narrative fidelity.

On StarWaves, we don't interact directly with the sound data as on a standard sound synthesis tool but we design the architecture of the scene which will compose the sonic organization through the spatial interactions and let perform this on a timeline. Here, the sound has no separate existence from space, time and motion.

## **TIME IS THE MIND OF STARWAVES.**

We can create alternate scene designs and store them as individual states. StarWaves makes it possible morphing gradually between these visible and audible alternates. The dimension of time will play us perceptual tricks during these transitions. As the time invades the audience mind, it develops the illusion of a theater of audiovisual forms that move in space.

## **WHY IN "SPACE" ?**

It is merely the very "space" which is an abstraction from our habitat and brings us to the non-gravitational environment of the universe, where there is no sound ! It is this emptiness which provokes the sensual penetration into it, and the sound is a perfect narrative medium to do this.

"NASA's next urgent mission should be to send good poets into space so they can describe what it's really like." Shannon Hale.

As the architects of these virtual scenes and themes of StarWaves, we feel the privilege of describing it with sound design and without doubt there is the poetry within.

## WHY STARWAVES ?

The name “StarWaves” comes from the sense of wonder and fantasy which has stimulated us during the project development and by witnessing its immense possibilities waiting us to discover.

As we keep looking for new frontiers in space and get fascinated with its new discoveries and the use of newest technology, we also need to investigate new instruments for sound design and new possibilities for musical creation. Analog synthesizers, keyboards and such equipment of the past were the cradle of electronic music, but one cannot live and progress in a cradle forever. Because I do believe that the future lying ahead is more prolonged than this past.

With current technological advances, we are limited only by our imagination and our will to create. And there are singularities on the timeline of advances in computer hardware and software applications. Today, StarWaves sees the daylight at one of them as it combines multiple disciplines and design elements powerfully and roll them into one high fidelity application.

sonicPlanet will continue to produce at great pace the next level of computer software for sound design, combining science, art and technology.

Dr. Sinan Bökesoy



## PRELIMINARY NOTES

- StarWaves window can run in full screen mode ( automatically full screen on iOS ). You can use the **Command + f** key combination and come back and go to full screen mode again.
- You can make the scene view of StartWaves full screen with the **Shift + u** key combination. Or you can double tap on the scene view to make it full screen ( both on OSX and iOS versions ).
- It is recommended to use a recent computer / iPad hardware ( iPadPro 2019 at minimum ) to run StarWaves.
- On iOS, StarWaves needs to be operating on the foreground always, otherwise its operation will freeze.
- Sometimes presets need a few seconds to load the sample materials and install the parameters. Performing the preset during this process may interrupt audio.

## FILE ACCESS / OPERATIONS ON STARWAVES

StarWaves apps are sandboxed , this is a necessity of AppStores ( both Apple and MicroSoft ) . The app data files will be put in special locations and the app itself cannot access the files outside of its app data directory.

Therefore once you locate the data files , you just need to do ordinary import / export operations to the relevant folders, such as the Presets folder or the Samples folder.

**Access on iOS :** On iOS please use the Files app of Apple to directly access the StarWaves app data folders. You can easily access them for any import/export and backup operation.

**Access on OSX :** On OSX, the StarWaves data folders will be installed by the system into following location.

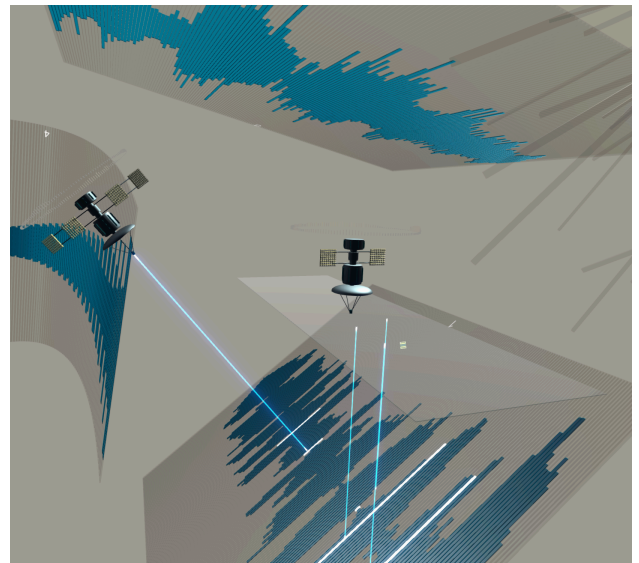
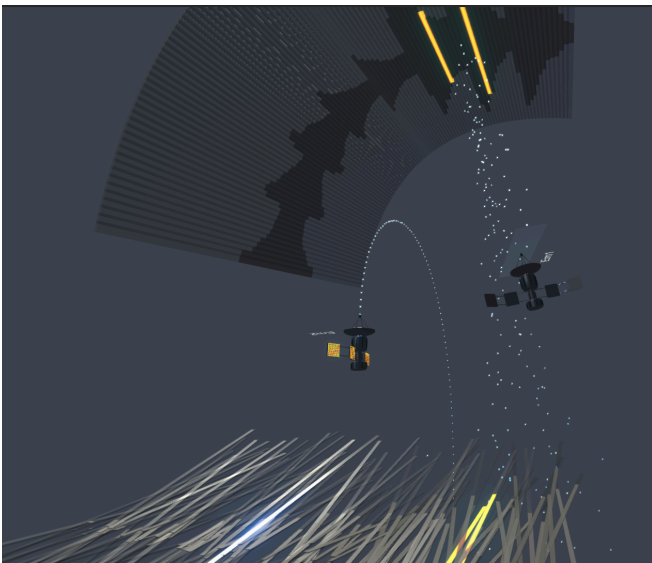
/Users/(yourusername)/Library/Containers/StarWaves/Data/Library/Application Support/com.sonicPlanet.StarWaves/

All the app data directories are found in there. You can create a shortcut of this location and easily access for any import/export and backup operation.

## THE STRUCTURE OF STARWAVES

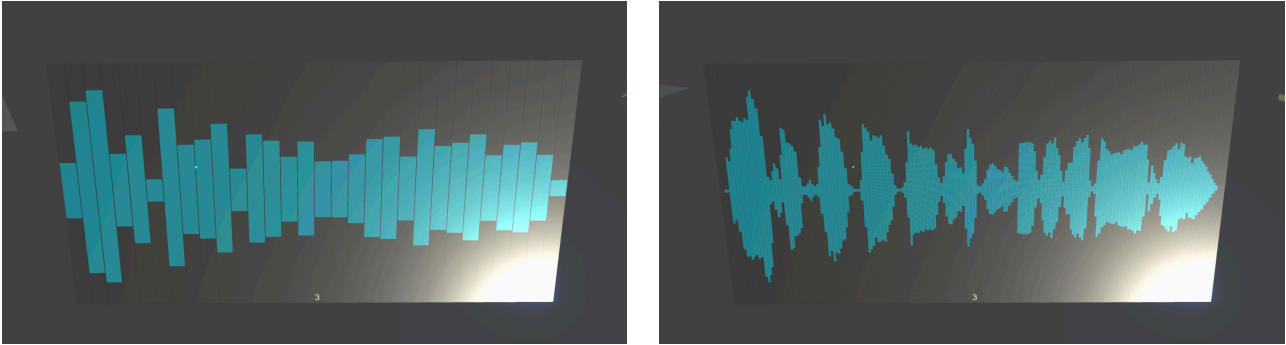
StarWaves includes 2 “**Emitters**” which generates particles or beams and 4 “**Platforms**” which hold the digital sample data and relevant synthesis engine to be triggered by these particles / beams.

The Emitters emit either particles or a direct laser beam and basically we expect them to hit the platforms triggering their waveform content. Below you can see both cases where on the left, the emitters create particles and on the right, the beams.

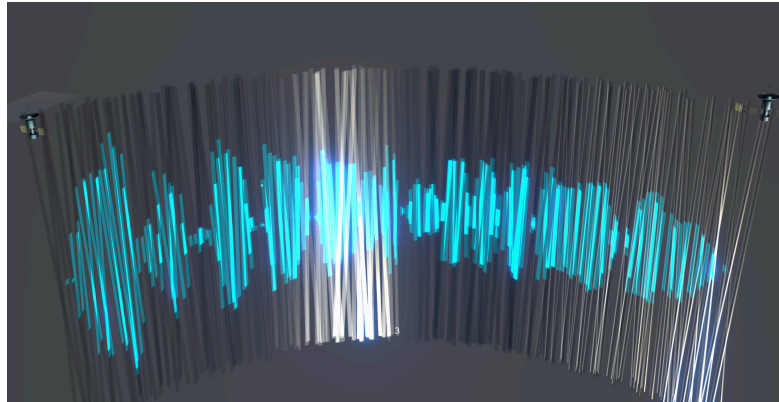


The Platforms contain the digital sample data and they represent this data segmented with 3D rectangular blocks. Once the sample data is loaded to a platform it divides it with a defined quantity of blocks by the user. (It can be between 1 - 250 ). The time each block receives a particle or a laser beam from an emitter, it will respond by flashing a color and playing the synth voice depending on its sample content and synthesis parameters defined by the hitting emitter’s sound engine. The hit point and its angle ( along the vertical and horizontal axis of each block ) defines also some parameters to be used for the sound synthesis process.

Below you see the same platform holding a sample data. On the left, it is segmented and divided with 30 blocks and on the right, with 214 blocks. Each block object represents the starting and ending point of the sample data on the timeline. The visualization however represents the RMS<sup>1</sup> value of the sample data in that block region.



Below you see again the same platform but bended with a parabolic curve and the blocks are a bit displaced with a random factor each. StarWaves has numerous tools handling the 3D position and rotation of platform blocks and therefore effecting directly the emitter hits which will be used for the sound synthesis process.



The emitters particles can be made bouncing from the platforms so that they might hit other platforms on their course. Also the laser beam can be reflected from the surface of the platform and hit another platform. ( A laser beam can hit 3 platforms at the same time , 1 direct and 2 reflected. )

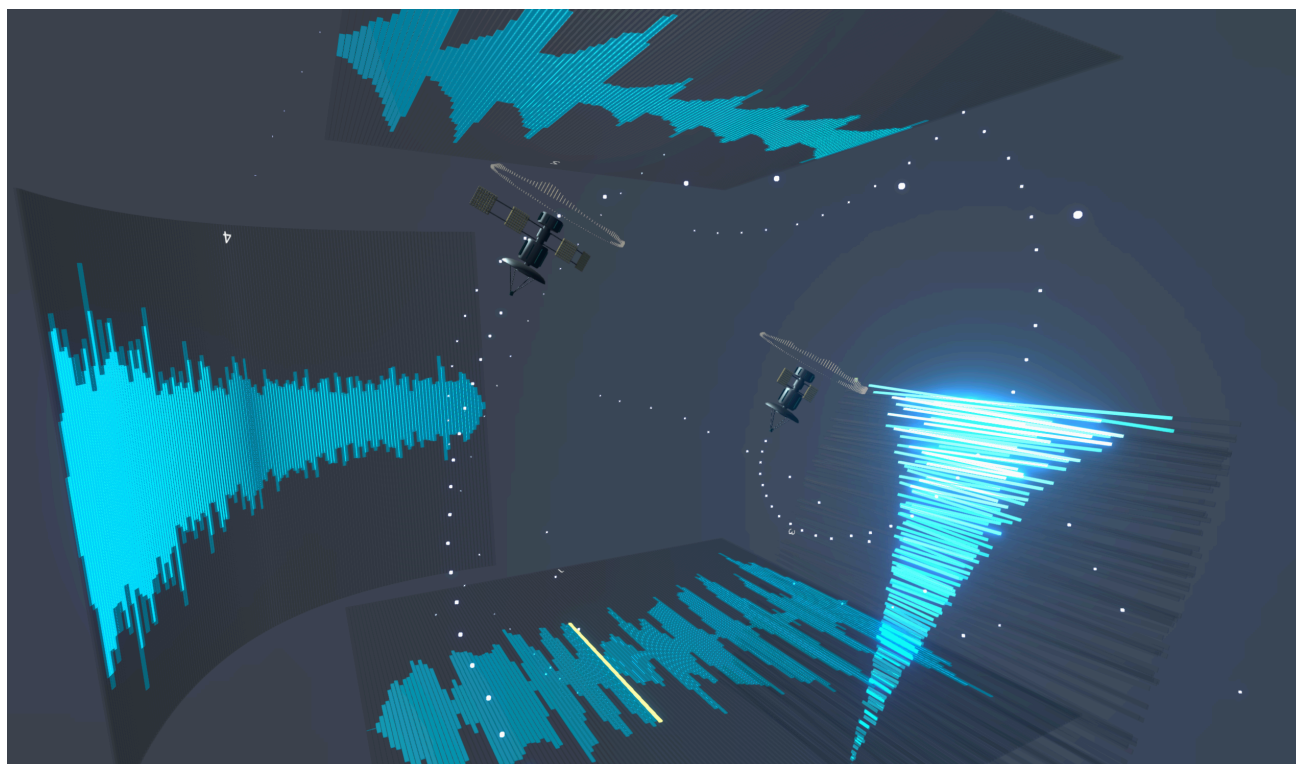
The emitter can send particles with variable rate / speed / duration. Also the emitter can send 3 laser beams at the same time with variable starting points.

---

<sup>1</sup> RMS : Root Mean Square of a sample block data gives the average power of the signal in that block time range.

In physics, light can be modeled by particles, moving in light speed and a laser beam is nothing else than a very focused and directed beam of light. So the dynamics of the StarWaves emitters are built with this inspiration.

The emitters and platforms constitute the space-atmospheric, non-gravitational 3D scene of the StarWaves. It is an interaction space where the dynamics of physical movement, kinematics which result to an audio-visual experience to be seen and heard in a direct relationship.

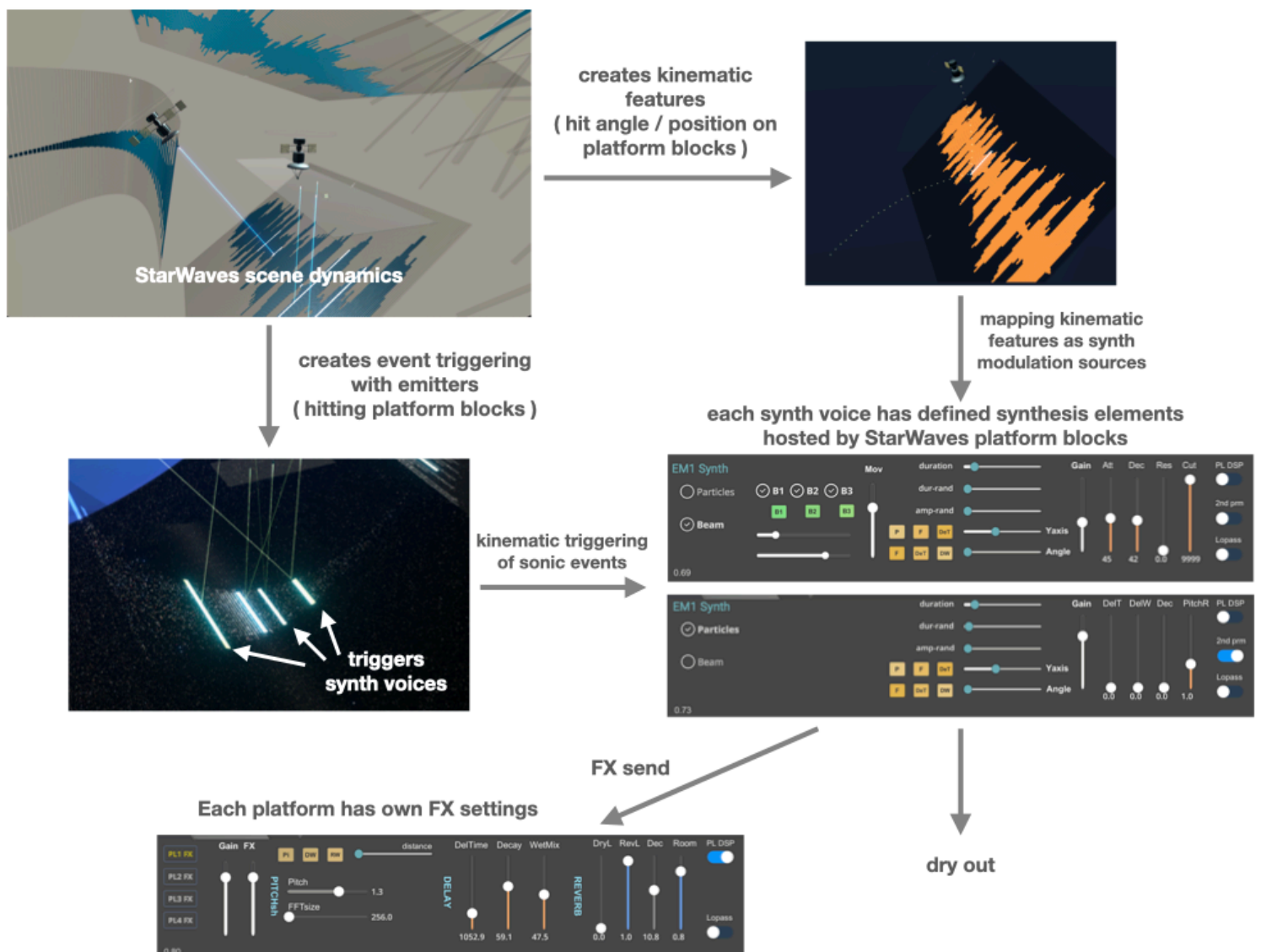


## THE FLOW CHART OF STARWAVES

The motor of StarWaves is the kinematic interaction between the emitters and the platforms. The platform blocks hold the sample data. They are representing the sound object as formed with concrete 3D blocks.

The blocks are triggered with emitter particles or beams when they collide with the platform blocks surface, each representing a synth voice.

Each collision event creates a synth voice trigger and its kinematic features serve also as modulation data for the synth engine of each voice.

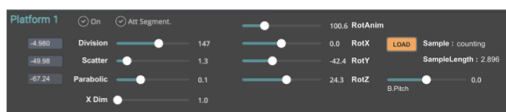


Each platform controls a dedicated FX section ( pitchshifter, delay, reverb ) where the platform block sound output can be sent for further processing.

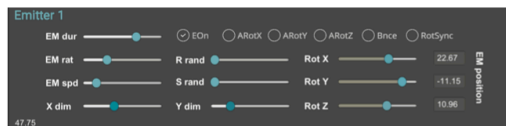
The platform and emitter control parameters let the user define their behavior, content and spatial orientation.

The FX section of each platform can use also some kinematic features to map to effect parameters such as the distance of the emitter from the platform.

The FX section output and Dry output from each platform combines the sonic output of StarWaves.



Platform parameters

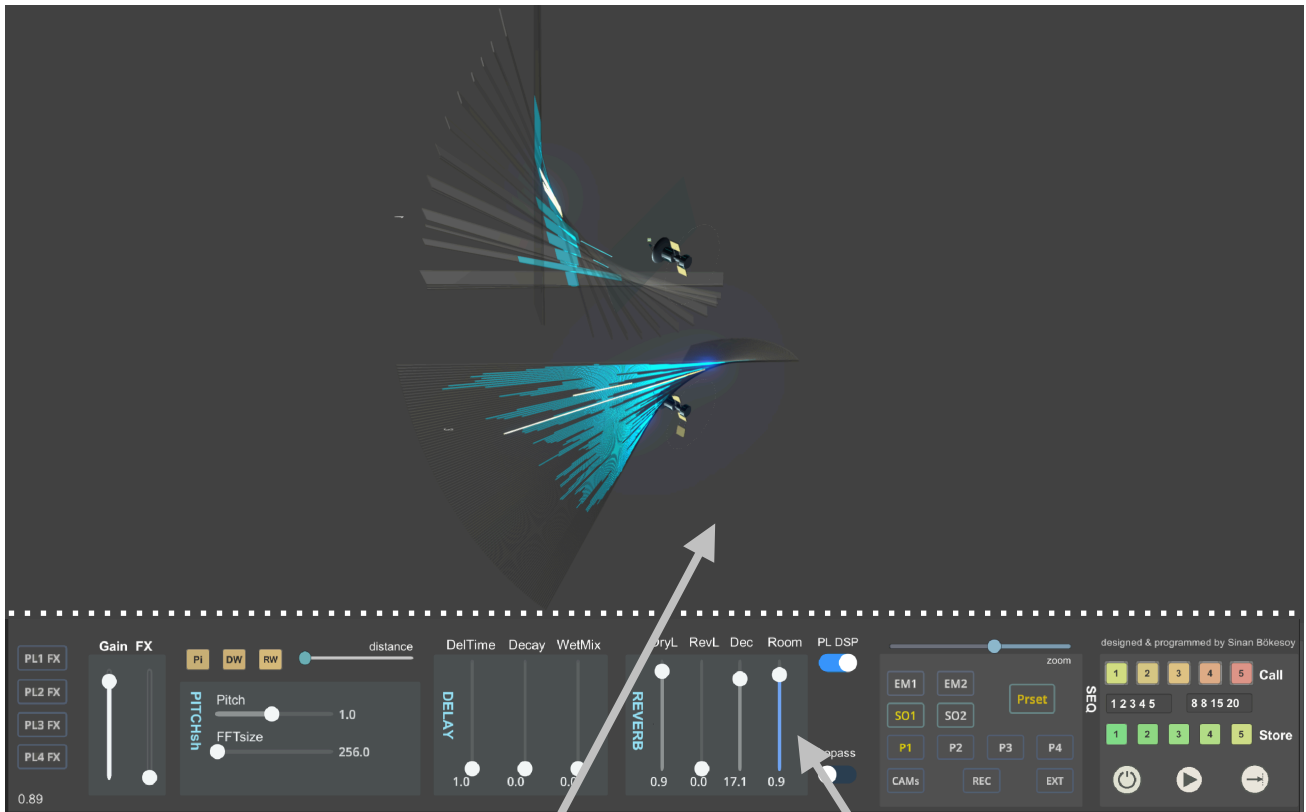


Emitter parameters



## THE CONTROL PANELS OF STARWAVES

The StarWaves control layout is minimalistic and is made reminiscent of retro-style space ship panels. There is a main panel and sub panels revealed according to the actions on the main panel. Below is the app opening panel state of StarWaves.



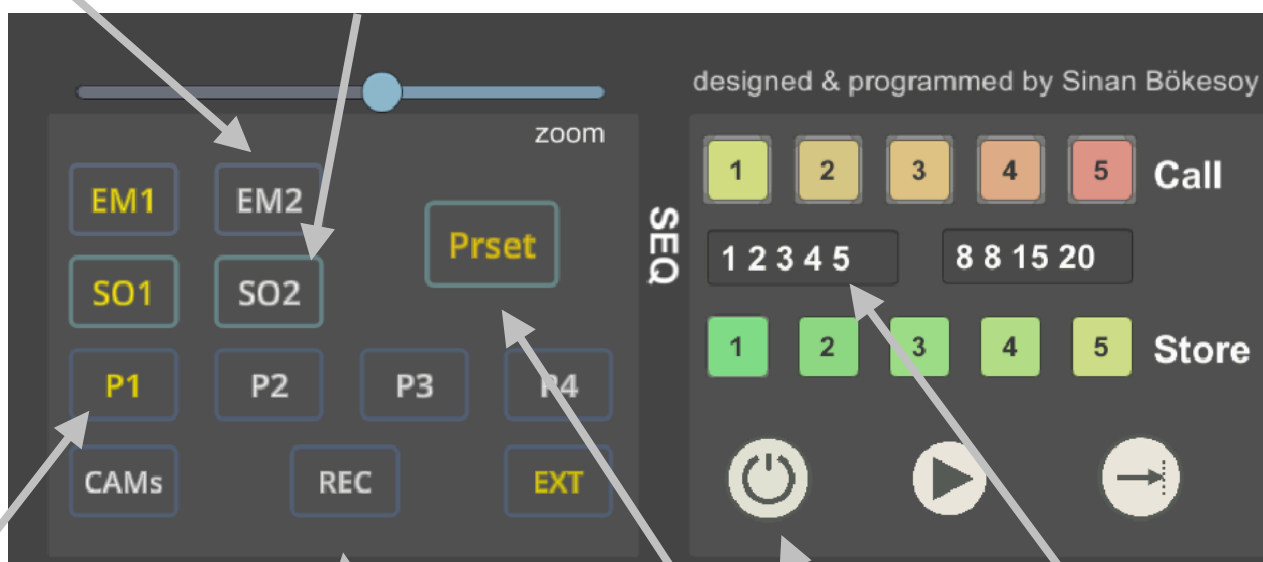
3D Scene area; you can interact with the scene objects directly.

Panel area; all control actions are made on this panel.



Emitter control panel

Sound engine panel for the emitter



Platform control panel selection

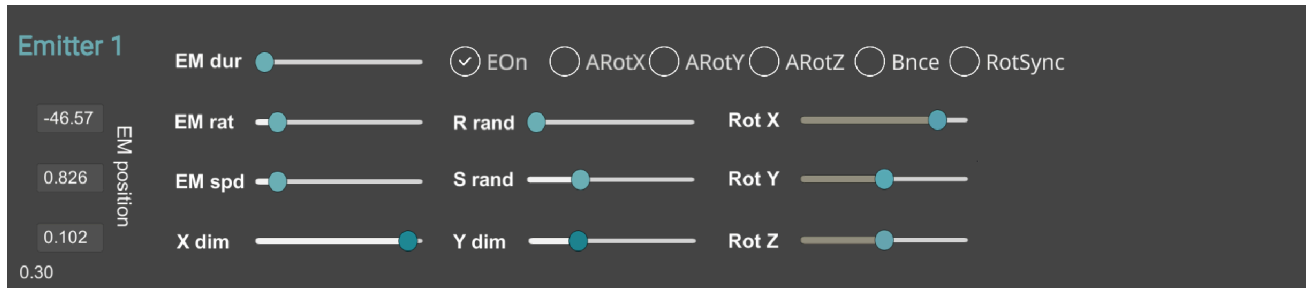
Preset panel

State sequence area

Sound output record action

Turn audio on / off

## THE EMITTER PANEL



The Emitter panel defines the behavior of the particle emitter and also the 3D spatial definition of the emitter itself. On the scene, Emitter 1 has always blue solar panels for you to easily distinguish it.

**EM dur** : This value defines the life span of the particles , if too short they might be dead even without hitting any object.

**EM rat** : Emitter rate defines the density of the particles emitted in one second.

**EM spd** : Emitter speed defines the speed / how fast the particles travel in space.

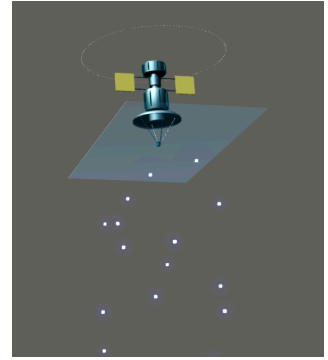
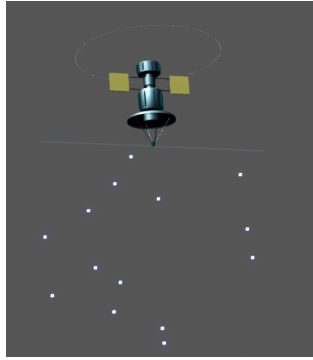
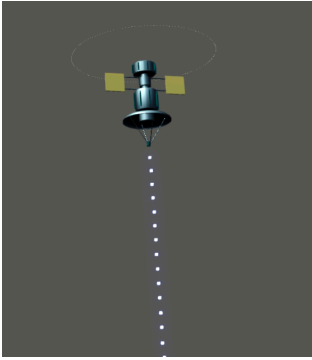
**R rand**: Emitter rate randomization factor

**S rand**: Emitter speed randomization factor.

**X dim** : Normally the particles are emitted from a single point ( the tip of the emitter satellite ). With this setting, we can distribute the particle starting points along the emitter x-axis line randomly ( the length of the line is set basically with the X dim value ).

**Y dim** : Normally the particles are emitted from a single point ( the tip of the emitter satellite ). With this setting, we can distribute the particle starting points along the emitter Y-axis line randomly ( the length of the line is set basically with the Y dim value ).

When using both the X dim and Y dim settings together, we can define a 2D plane which distributes the particle start at random points on this plane. Below you can see the example use of these settings.



**Rot X** : You can rotate the emitter satellite around its x-axis.

**Rot Y** : You can rotate the emitter satellite around its y-axis.

**Rot Z** : You can rotate the emitter satellite around its z-axis.

To move the Emitter satellite you can alternately grab its emitter tip and move inside the scene view. The altered coordinates will be exactly shown on the EM position fields. For precise editing the coordinate values, you can type in the value by clicking on the relevant field.

As you see, there are 6 switch buttons on the top line of the EM control panel. Let's go through their functions.

**Eon**: Starts and stops the emitter activity.

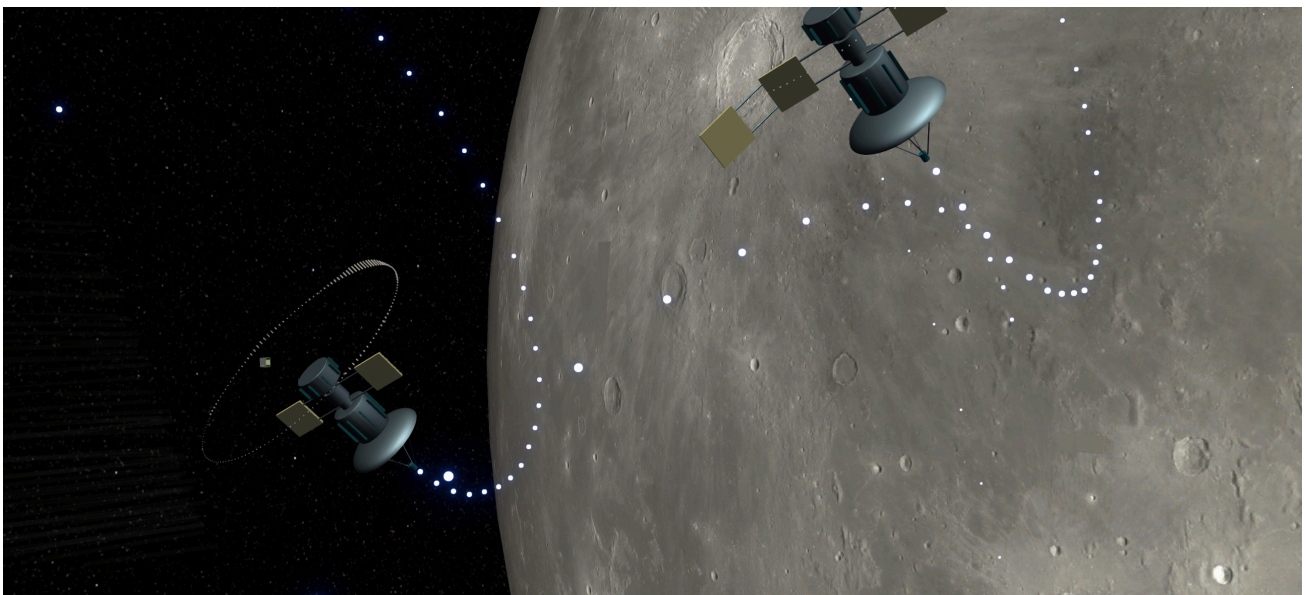
**ARotX** : Continuously rotates the emitter around its x-axis. The speed of this rotation will be set now with *Rot X* slider.

**ARotY** : Continuously rotates the emitter around its y-axis. The speed of this rotation will be set now with *Rot Y* slider.

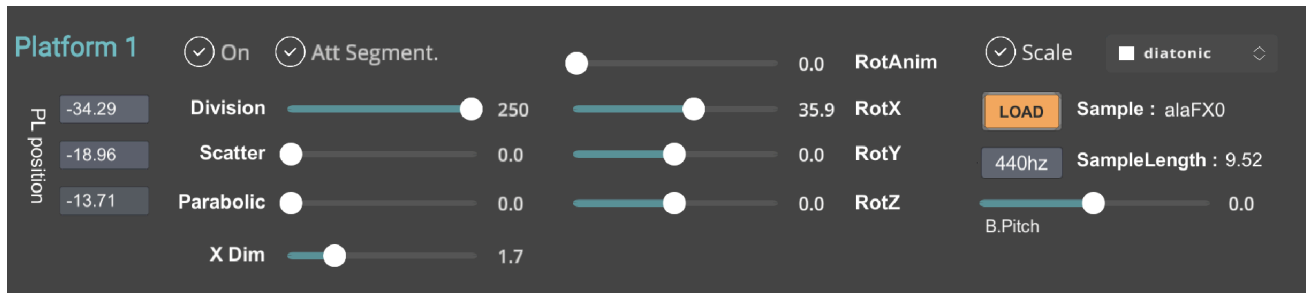
**ARotZ :** Continuously rotates the emitter around its z-axis. The speed of this rotation will be set now with *Rot Z* slider.

**Bnce:** When turned on, the particles will bounce on the surface which they hit, otherwise they will terminate there.

**RotSync :** When turned on, the second emitter satellite will sync to the first emitter so that it becomes its twin. Beautiful motion choreography can be established on this mode and playing with the rotation dynamics.



## THE PLATFORM PANEL



You can grab and move each platform with the mouse ( or tap and drag it on the iPad ) on the scene view. The coordinates will show on the position field boxes. Alternately you can type in the exact coordinates by clicking on these field boxes.

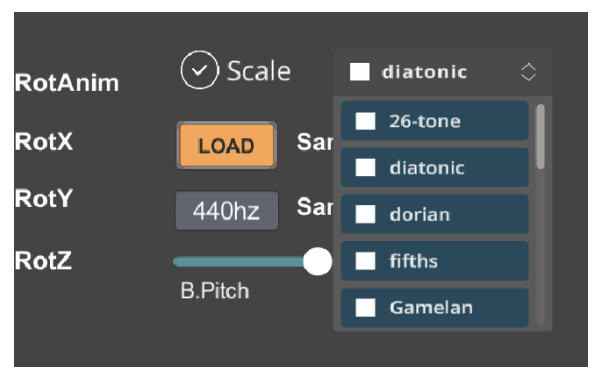
A platform needs a sound sample content. In order to load your sample, click on the **LOAD** button. Only mono **.wav** files format is supported ( ensuring cross-platform compatibility ) Please use sample files with less then 22sec. duration.

**Attention:** Please use only mono samples, using stereo samples is pointless.

When you load a sample successfully, its name and length will be updated. Also you will see the platform visual updated projecting your sample data.

**Scale :** The Emitters perform the sample content of each platform by using particles or beams scanning over the platform surface. We can a the pitch modulation by using the Y-axis of the platform, so depending where a particle or beam hit the platform surface along the Y-axis , the pitch can change proportionally ( the settings are explained on the next chapter ( Synth Engine panel ). By activating the “Scale” button, we can choose a scale to quantize these pitch values so that the distributed pitch values can be part of a chosen scale on its pop-up menu.

**B.Pitch field :** Here we type in the base frequency value of the sample. For instance if it is a piano sample on middle A key, then the frequency would be 440hz. As soon as you



change this value, a new scale distribution will be calculated by taking this new base freq. value.

**B.Pitch slider** : This slider changes the base freq continuously by taking the above typed in value as its middle position reference value.

**Division** : This defines the number of blocks which represent the sample data on the platform. As explained at the beginning section of the manual, each block holds a portion of sample data with a duration and specific starting and ending times on the sample waveform time line. The more blocks, the shorter their duration. The visual on the platform / each block represents the average RMS value of this block sample data. When a particle or laser beam hits a block, it will get triggered and play its content as a synth voice in accordance with the settings of the emitter sound engine.

**Scatter** : This function will distort the platform by changing each blocks rotation and position randomly. The slider defines the degree of this distortion.

**Parabolic** : This function bends the platform with a parabolic function. Likewise it behaves like a lens when used with emitter beams and is a lot of fun to experiment with.

**X dim** : This value of this slider stretches the length of the platform.

**Rot X/Y/Z** : You can rotate the platform around its relevant axis. When you move a rotation slider a helper axis visual will be drawn on the platform.

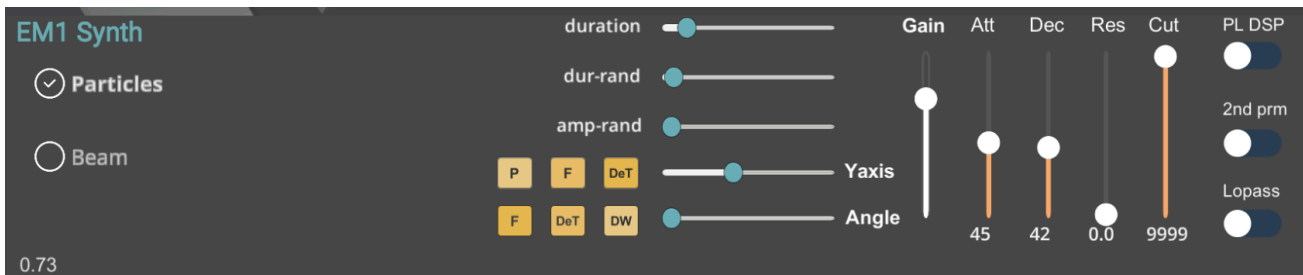
**RotAnim** : This spins each block along the x-axis of the platform gradually depending the value of this slider.

Now about the two switch buttons on the top of the Platform panel;

**On** : This makes the platform visible or not on the scene view.

**Att Segment** : This changes the block segmentation from linear to attack sensitive mode. Likewise, the sound sample of the platform will be analyzed and segmented according the attack content. This is great for using rhythmic groove-style sonic content where platform block start and stop points can be matched at beat points by using this mode.

## THE SYNTH ENGINE PANEL



Each emitter has its own synth engine panel as their synth engines are independent. And the synth engine panel has itself sub-panels. Let's cover them here.

On the left part of the panel, you can choose the emitter behavior. As mentioned before the emitter can send particles ( of which parameters are set mainly on the Emitter Panel itself ) or the emitter can produce beams. Whichever you choose, it will share the same signal processing chain.

When a particle or beam hits a platform block, it triggers a sonic event which starts playing the sample data hold by that block and its duration is defined by the **duration slider** of the Synth Panel. It can be in the range from 1ms to a second.

Additionally you can randomize this duration with the **dur-rand slider** setting and also randomize the event amplitude with the **amp-rand slider** setting.

Next, on the right side you will see 5 vertical sliders;

**Gain** : This controls the gain of the sonic event.

**Att** : This applies an attack ( fade in ) as a gain envelope. The setting is related to the attack time. ( The beam mode has a fixed attack and decay internal setting. )

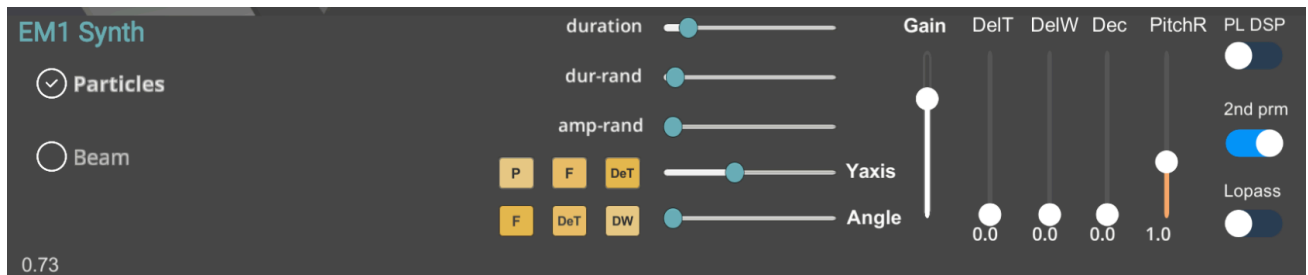
**Dec** : This applies a decay ( fade out ) as a gain envelope. The setting is related to the decay time.

**Cut** : This sets cutoff frequency of the filter which is applied to this sonic event. We can choose low pass or high pass filter.

**Res** : This sets the resonance value of the filter, emphasizing the frequencies around the filter cutoff.

**Lopass / Hipass** : This switch defines the type of the filter.

The switch button called “**2nd Param**” will open another small sub-panel with additional parameters.

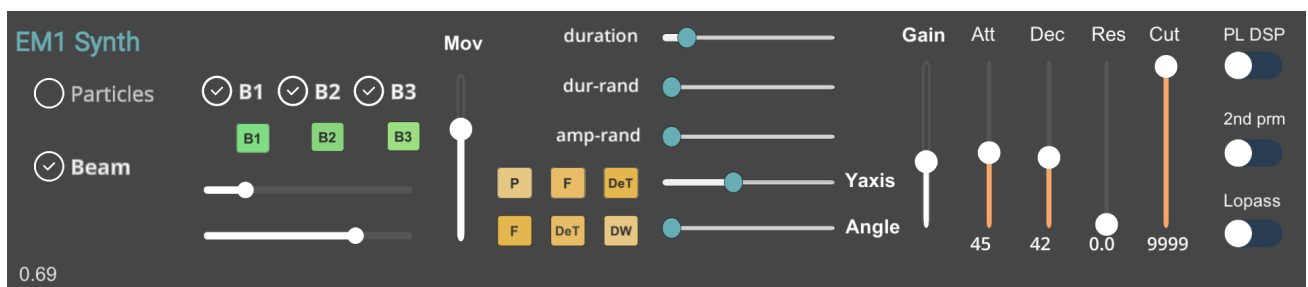


**DelT** : This sets the delay time for a mono delay processing which is part of the emitter synthesis engine and applied to all triggered sound events by the emitter. Very low settings of the delay time with proper decay setting can create comb filter effects.

**DelW** : This is the wet balance for the delay effect.

**Dec** : This is the delay decay setting. Higher values will cause more feedback.

When the emitter is on Beam mode, you will see the additional beam activation and positioning parameters on the left side.



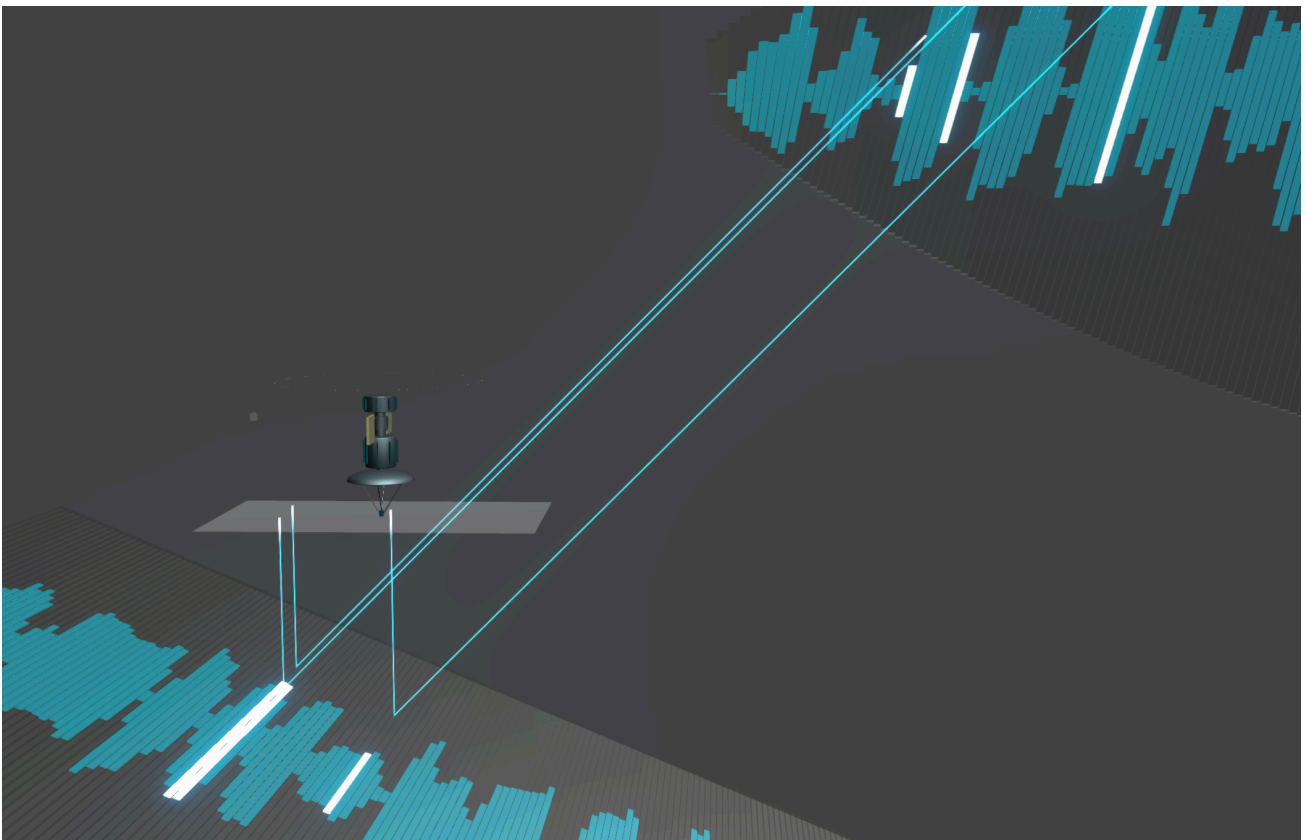
As mentioned before there can be 3 Beams sent at the same time in parallel from an emitter. The switches **B1**, **B2** and **B3** do activate those beams.

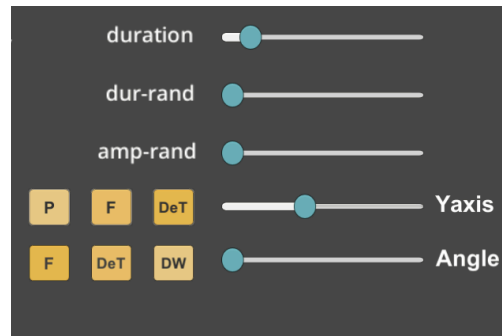


If you have set an X dimension / Y dimension for the emitter to expand the distribution over a plane surface, then you can define the starting positions of the beams with the two horizontal sliders within the X dimension / Y dimension ranges.

The top slider serves the x starting position of the beam and the bottom slider the y starting position of the beam. First you have to select the beam with the green beam selector buttons and then those sliders will serve for that selected beam.

The **Move** slider defines the degree of randomness of the beam position change among the X / Y dimension ranges. Hence, the beams will continuously move and reveal a fantastic motion element. Below you see an example case where an emitter is sending 3 beams, they are each hitting relevant platform blocks underneath and then also do bounce with an angle and hitting the other platform blocks on the top.



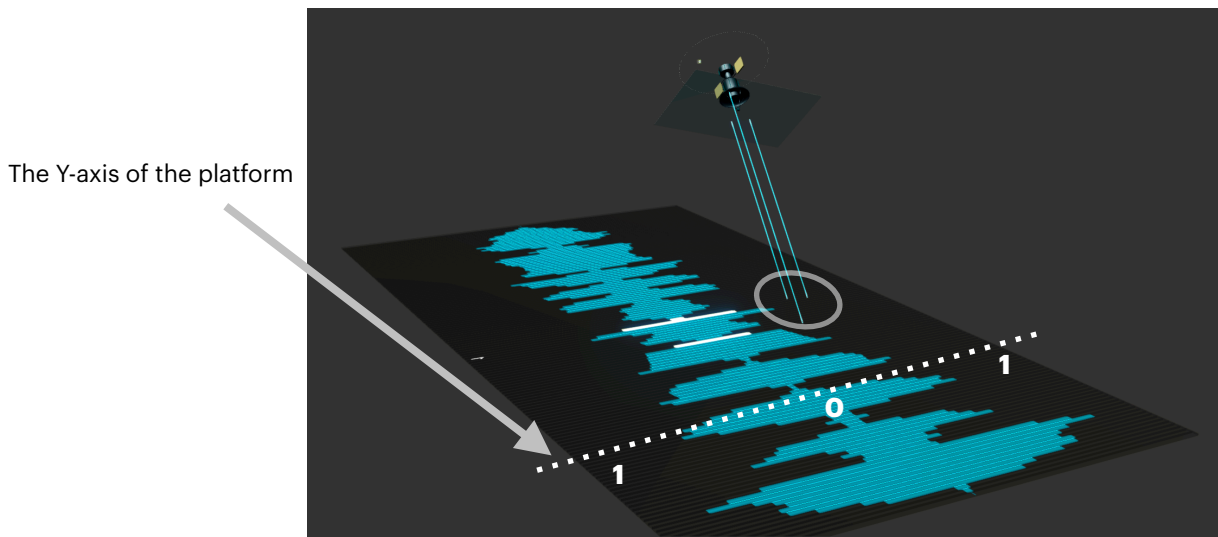


This section of the synth voice panel offers us a new paradigm in modulation design for sound synthesis, by extracting the kinematic aspects of the StarWaves scene and map them to signal processing tools. These kinematic features are derived solely from the interaction of the emitter and the platform which it is hitting.

We do focus on two kinematic features of the emitter and platform interaction.

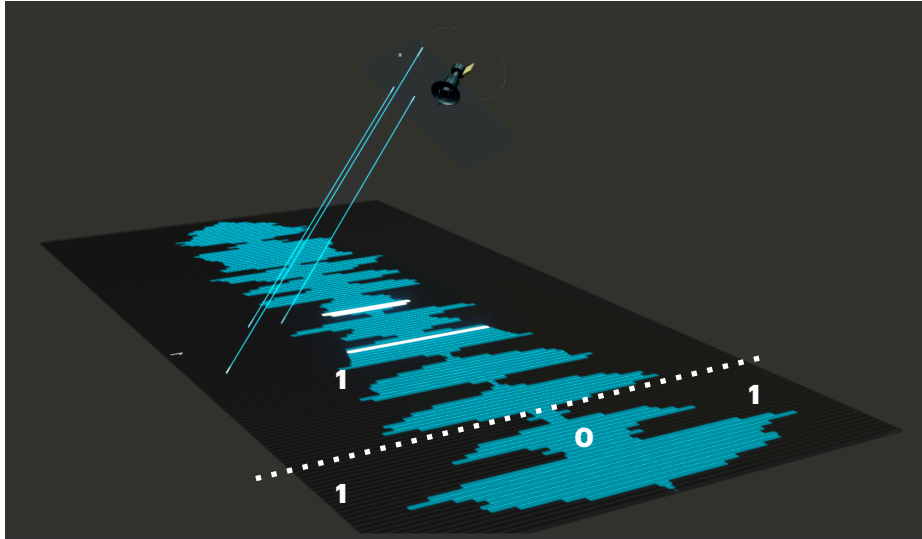
1. The Y-axis position of the hitting particle / beam on the platform surface.
2. The hit angle between the particle / beam and the surface.

Below, the relevant shots will clarify these aspects.

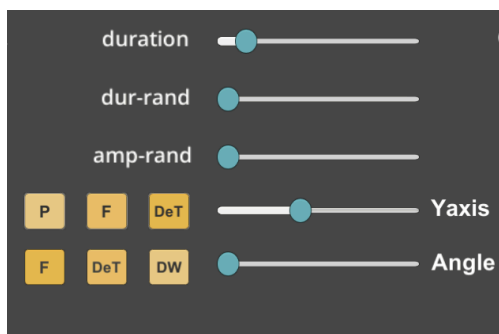


On the scene shot above, an emitter is rotating and entering the platform area and hitting it with 3 beams, and as you see the relevant blocks are lit up. Also the y-axis of this platform has been drawn so that you can see it clearly.

We have designed the interaction so that when the hit happens towards the center of the y-axis of the platform, it will create minimum effect but towards the top and bottom edges of the platform it will produce the maximum effect. And above, the emitter beams hit the platform surface near to the top edge so that the modulation value will be high.



Since the emitter rotation is continuous, the beams will traverse along the Y-axis and on the shot above we see that one beam is close to the center and another one is closer to the edge producing different modulation effects.



To map this Y-axis hit modulation, you can set with the **Y-axis slider** for of the following sonic processing destination.

**P** for Pitch, **F** for Filter Cutoff, **DeT** for Delay Time.

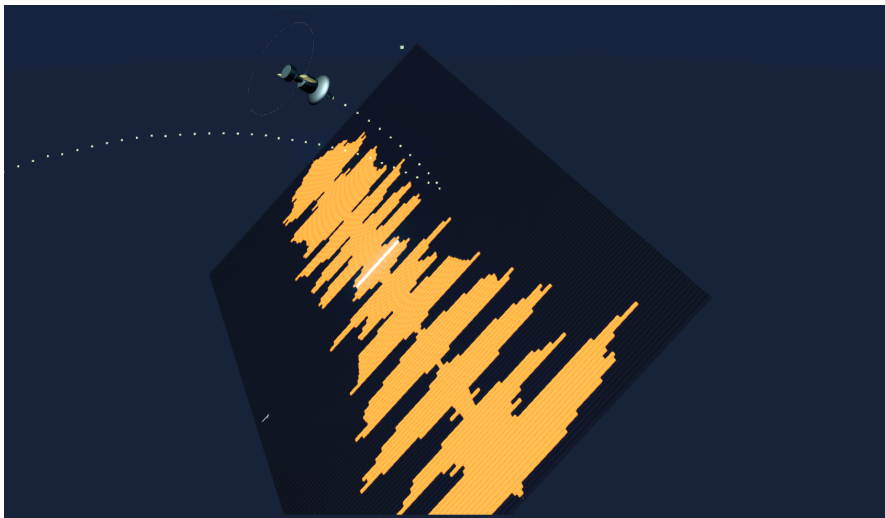
For instance, press the **P** button and now the **Yaxis** slider will serve to set the modulation depth effecting the *pitch* parameter. When it is at the center, there will be no modulation. When the slider is set to zero, then when the particles / beams hit the surface close to the edges , the pitch will decrease one octave.

And when they hit close to the center of the platform, the pitch value won't change. Alternately, if you set the **Yaxis slider** value at maximum, the pitch value will be one octave higher.

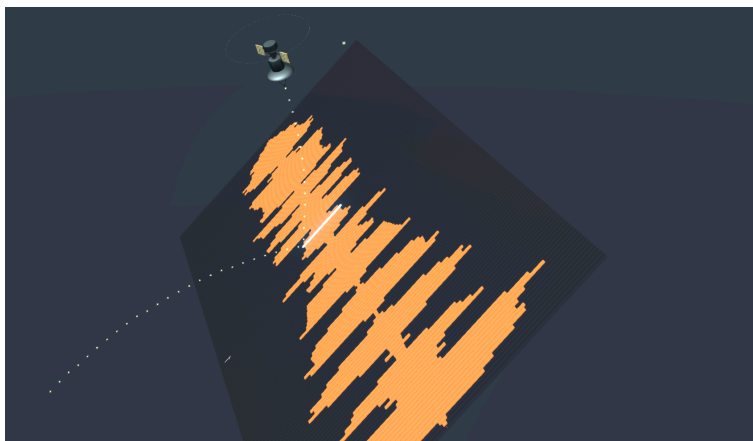
All these pitch changes will happen continuously depending on the hit position along the y-axis of the platform.

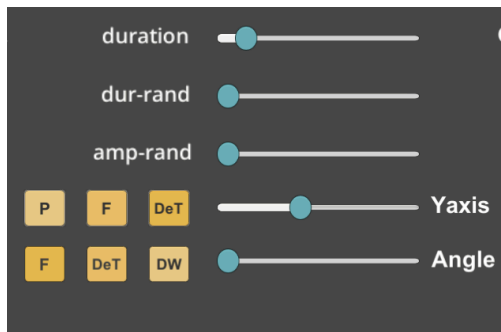
A similar logic works for the Filter Cutoff and Delay Time parameters as well. You can modulate them equally with the particle/beam hit dynamics along the Y-axis of the platform surface.

The second kinematic feature we do use for modulation on the synthesis engine is the hit angle between the particle / beam and the surface. On the below example, you see that the particles are hitting the surface nearly with 90 degrees.



And here the particles are hitting the surface with angle close to 45 degrees. The less the angle, the more the modulation effect will be.



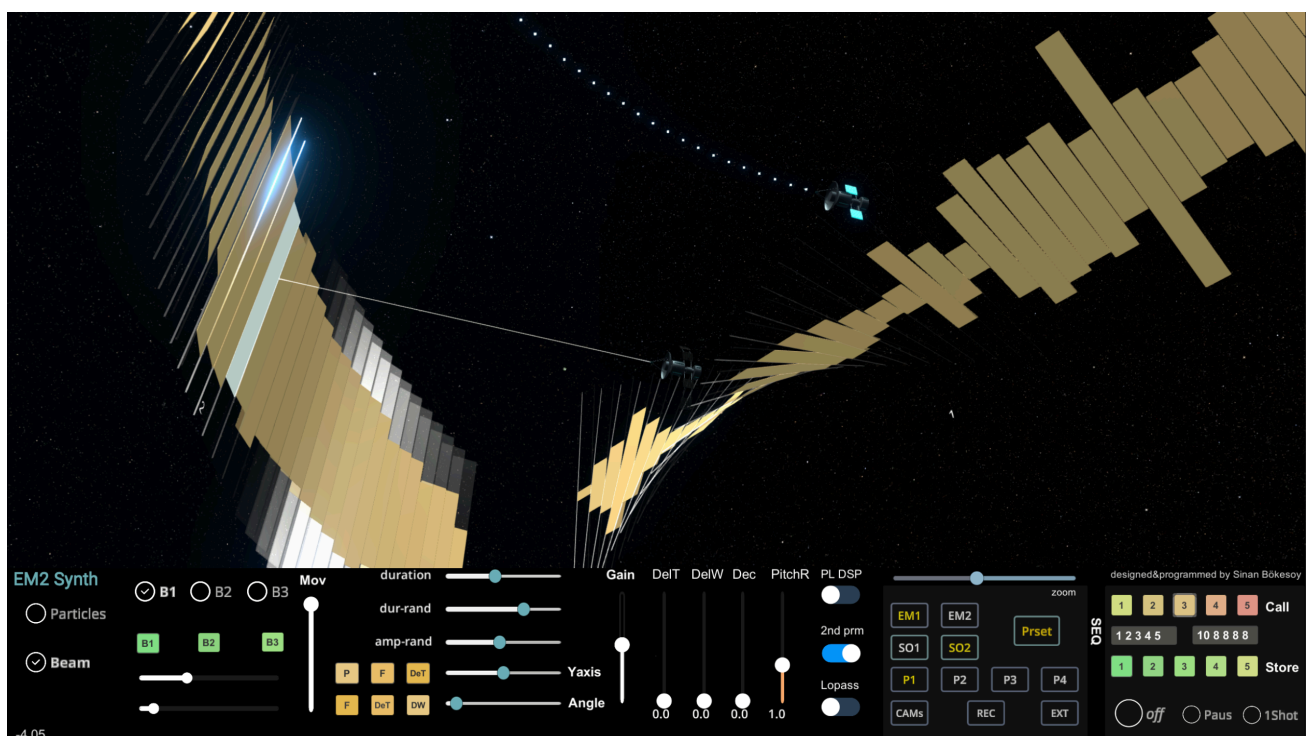


In order to map the angle modulation, you can set with the Angle slider for of the following sonic processing destinations :

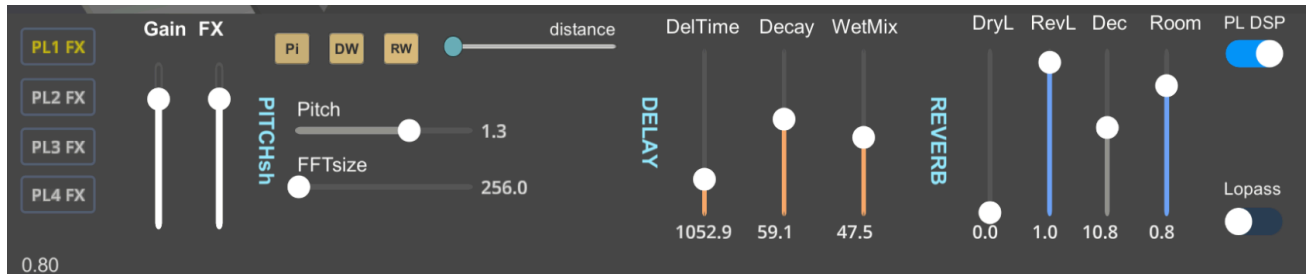
**F** for Filter Cutoff, **DeT** for Delay Time, **DW** for Wet balance for the Delay effect.

For instance, press the **F** button and now the **Angle** slider will serve to set the modulation depth effecting the *Filter cutoff* parameter. The value of the slider will act as the depth of the modulation which is produced by the hit angle value. A similar logic works for the Delay Time and Delay Wet balance parameters as well. You can modulate them equally with the particle/beam hit angle value on the platform surface.

StarWave scenes can exhibit a complex and continuous kinematic interaction. One event can trigger other happenings as a network of connected events. The above modulation mapping will help to perceive these dynamics as sonic transformations.



## THE PLFX (PLATFORM FX) PANEL



Each synth voice can send its audio output to Platform FX section for further processing. Each of the 4 platforms on StarWaves offer their dedicated effects section. For example, an emitter can distribute in space its particles and they can hit all 4 platforms at different times. Each platform block they hit will be processed by that platform's effect section.

You can access the Platform FX panel by using the **PL DSP** switch on the synthesis engine panel. You will see the above sub-panel.

On the far-left side you see the **PL1 FX**, **PL2 FX**, **PL3 FX**, **PL4 FX** buttons. By pressing on them you can switch to each platform's effect settings. Let's explain these settings.

The **Gain** and **FX** vertical sliders control the dry signal out and processed FX signal level of the relevant platform effects section.

The platform effects are the following and established in a serial order, and each effect has its dedicated parameters.

*Pitch Shifter -> Delay processor -> Reverb processor.*

It is designed by keeping in mind adding beautiful shimmer effects on the platforms of StarWaves.

The **Pitch** and **FFTsize** horizontal sliders control the pitch shift amount and the FFT size being used for this process.

From the Pitch shifter output the signal goes to the delay effect. The **DelTime**, **Decay** and **WetMix** horizontal sliders are dedicated to the Delay processor. They set the delay time, the delay decay time ( feedback ) and the wet delay signal level of this delay effect.

From the delay processor, the signal goes to the Reverb processor. The **DryL**, **RevL**, **Dec** and **Room** vertical sliders are dedicated to the Reverb processor. They control the dry level entering to the reverb, the reverb level of reverb processor, the decay time of the reverb effect and the room reverb density of the reverb effect.

All these settings are unique to each StarWaves platform and you can morph these parameters between StarWaves scene states.



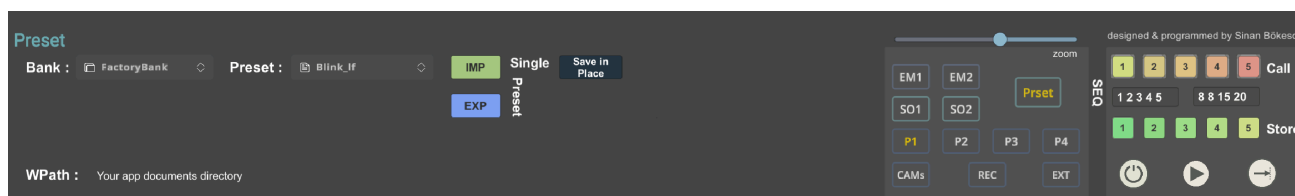
Like on the synthesis engine, we can map a kinematic parameter for modulating these effect parameters.

The feature we do focus here is the distance between the emitter and the platform which it's hitting with its particles or beams.

The **distance** slider will set the depth of this mapping for each of the parameters **Pi**, **DW**, **RW**. They are Pitch shift amount, delay wet signal amount and the reverb signal amount.

For an example, the distance between the platforms and the emitters can increase the reverb signal amount , when you set the distance slider for the **RW** button. This will wash out the distant hit events.

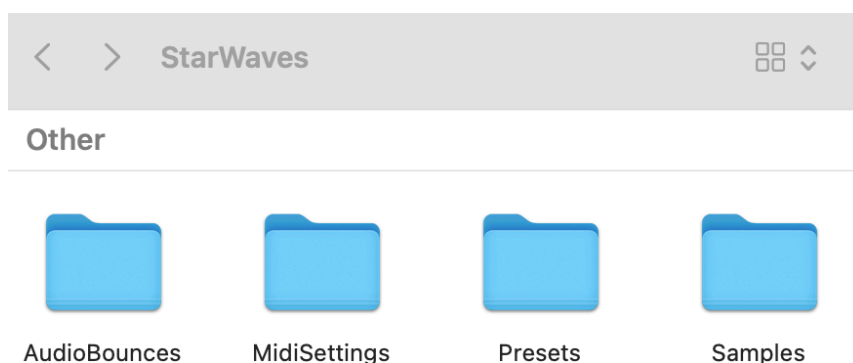
## THE PRESET PANEL



The preset panel can be reached directly by pressing the *Preset* button on the main panel.

StarWaves presets are grouped in banks and each bank contains 12 presets. Banks are merely folders in your StarWaves work directory. You can re-organize them at your will. Each preset comes with 5 scene states stored on the sequence panel. They are scene variations and can be called by pressing the relevant state recall buttons. ( or use the key shortcut Shift + state no )

Whenever you install the app and launch it first time, it will create its work directories inside the directory of *StarWaves*. Let us explain them now.



**Presets** : This folder contains the banks holding the presets of StarWaves. The Factory Bank and StarBank1 comes by default and hold 12 presets each and also there is a UserBank created by default.

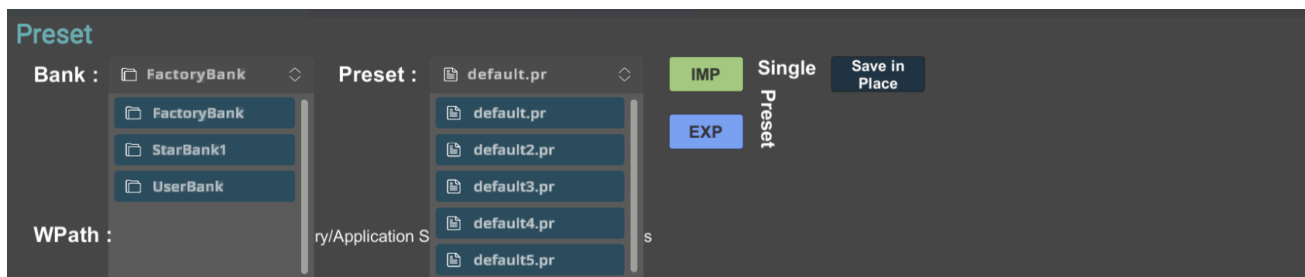


**Samples** : This folder holds the sample files used by the presets. Also when you use the *Import Samples* feature of the app, the imported sample will be located here for future use.

**MidiSettings** : This folder holds your midi settings presets ( Midi and OSC is an In App purchase add-on ) All your controller assignments can be packed as a midi setting preset.

**AudioBounces** : You can record the audio output of StarWaves by using the **REC** button on the main panel. With each successful record an automated wav. file will be created in this folder.

In order to load a preset, first select a bank by using the dropdown menu on the left. As soon as you select a bank, its preset list will be uploaded to the preset dropdown menu on the right. Here you can select a preset, and its content will be loaded automatically.



Alternately you can load a single preset by using the **IMP** button, which will lead you to a file browser so that you can select the preset.

Now that you have worked on a new preset and want to save it, use the **EXP** button. The file browser will open up, you give your preset a name and then click save.

If you just want to re-save the last loaded preset with your edit, you can use the **Save In Place** button. It will ask for your confirmation to replace the old preset in place.

## WHERE IS THE APP DIRECTORY LOCATED ?

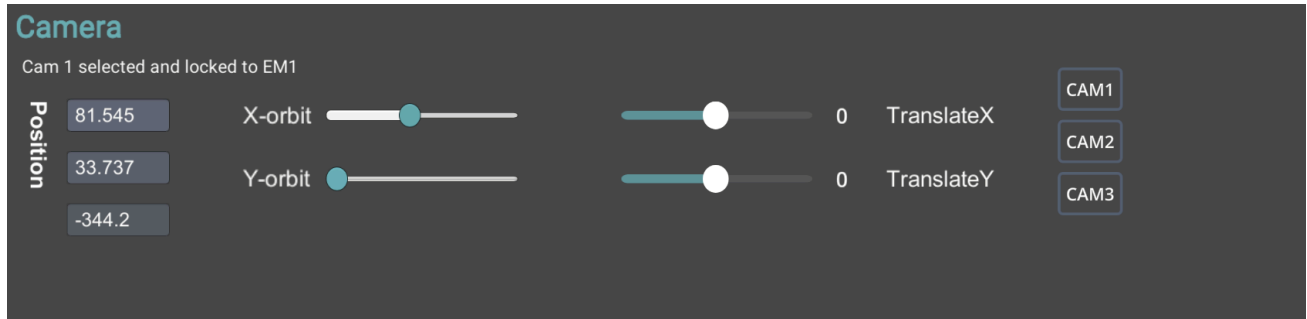
**WPath :** /Users/sinanbokesoy/Library/Containers/com.sonicPlanet.StarWaves/Data/Library/Application Support/com.sonicPlanet.StarWaves

Apple and MicroSoft has certain regulations for the apps sold on their AppStore. For instance, the app work files cannot be placed freely. Likewise, the app uses a certain path to put its folders/files for later use and for the access of the user. Not following these restrictions by the developer will lead a rejection of the app on the AppStore.

Please consult to the Files Management / Access section at the beginning of this manual for the location of the data folders for each system.

## THE CAMERA PANEL

You are looking to the StarWaves 3D scene through a camera. And it has its own settings panel.



One can directly use the mouse ( touch screen on the iPad ) and navigate / rotate the camera by clicking/dragging on the scene view area.

However by using the sliders on the Camera panel you can do the following camera modifications : You will see on the right, the 3 **CAM** buttons. They locate the Camera position.

**CAM1** positions the camera so that it faces and locks itself to Emitter1 **CAM2** positions the camera so that it faces and locks to Emitter2 and **CAM3** positions the camera so that it faces and locks to the center area from all objects on the StarWaves scene.

**X-orbit** : This slider rotates the camera horizontally around the center point which it is looking at. ( see above the camera positioning buttons )

**Y-orbit** : This slider rotates the camera vertically around the center point which it is looking at. ( see above the camera positioning buttons )

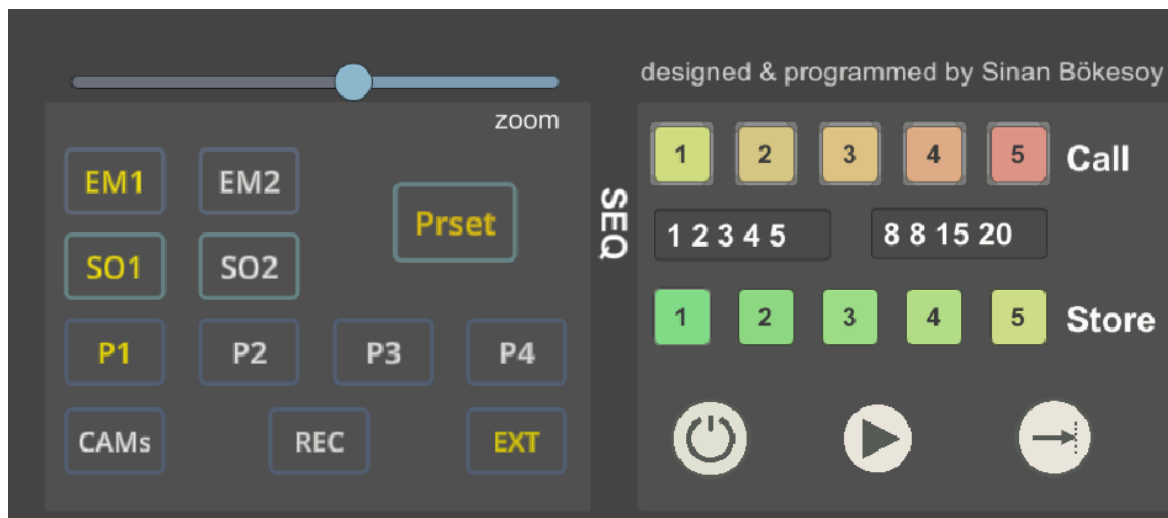
**Translate-X** : This translates the camera position to towards its left / right.

**Translate-Y** : This translates the camera position to towards its up / down.

You can also use the camera position fields on the left to type it the position coordinates manually if needed. On the main panel, you will see a camera **zoom slider**, which will actually change the distance of the camera from the point at which is it looking.

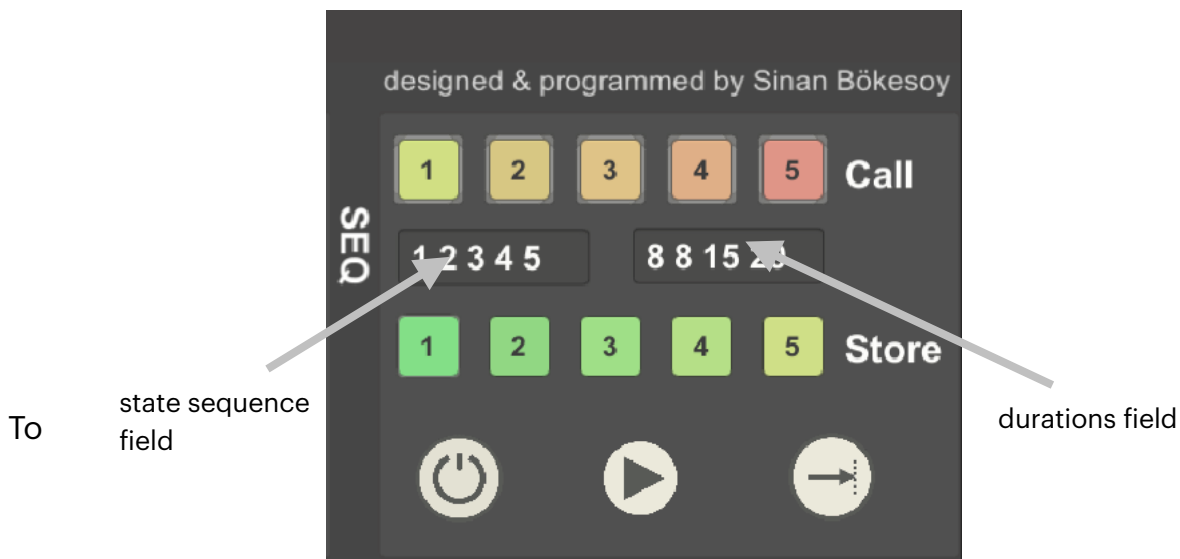
## THE SEQUENCE PANEL

The sequence panel is part of the main panel and displayed at all times. The idea is basically storing all scene parameters / panel parameters as individual states and call these states back with various options.



StarWaves offers 5 state slots where you can save 5 unique states of the scene. There are 5 numbered buttons dedicated to each of them, you can use the ones on the upper row to call back the saved states. And In order to store a scene state, double click on a button on the bottom row. ( the button surroundings will flash with a successful store )

All this is nice but how about a continuous transition from one state to another ? For such an audio-visual morphing process, we need to set the duration / how long it will take between states in transition.



To determine the transition durations between states, type on the *durations field* ( indicated above ). The first number will set the transition duration ( in seconds ) between state 1 and state 2. The second number will set the transition duration ( in seconds ) between state 2 and state 3. this will go on like that...

**Attn** : *Always leave a space between numbers and hit enter when you are done.*

On the example case above, the transition between states 1 and 2 is 3sec, between states 2 and 3 the duration is 4 sec and between state 3 and 4 , the duration is 2 sec. If there is no duration set for a certain state transition, the default duration value will be valid.

We can also automate the state transitions by typing its sequence on the *state sequence field*. On the example case above, the state transition sequence will start with state 1 and then move to state 2 and will end at state 3.

**Attn** : *Always leave a space between numbers and hit enter when you are done.*

In order to achieve that animation you have to start it with the switch at the bottom middle. When active, it will be on the **play mode** and when not, it will remain on the **pause mode**.

This will perform the defined state sequence. This can be a one shot performance or a looping one.

In order to loop a sequence of states, the sequence should not start and end with the same state.

All these scene states and sequence settings are part of a StarWaves preset.

On OSX , you can use the key combination **Shift + state no** to call a state. And use **Command + state no** to store your current scene on a state slot.

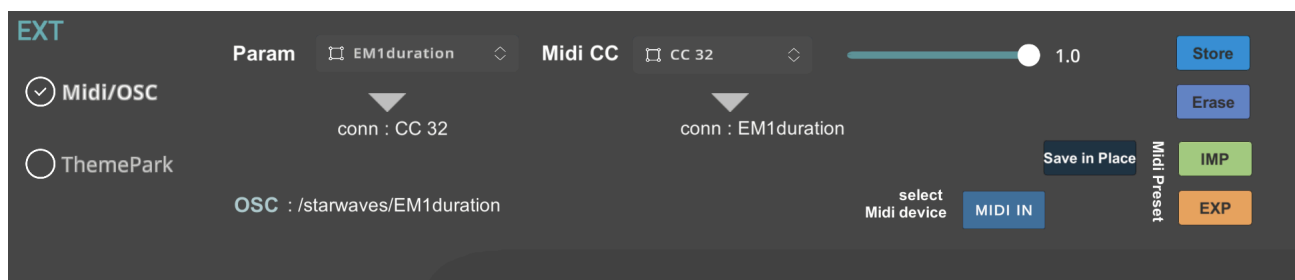
*Only continuous slider data can be morphed between states. Particle voice attack and decay values won't change between states.*

## THE EXTERNAL FEATURES PANEL

To access this panel, click on the **EXT** button on the main panel display.

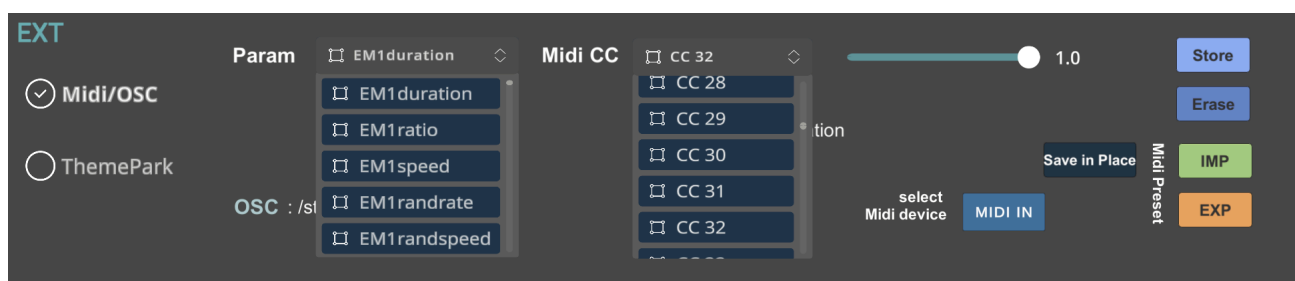
The External features panel exhibits the **MIDI / OSC settings** and the **Theme Park panel**, both are In App Purchase options which you can purchase on the AppStore for your StarWaves app.

### MIDI / OSC PANEL



The MIDI / OSC add-on panel will reveal itself automatically after the In App purchase process and restarting the app. All the parameters of StarWaves can be controlled now with a Midi CC controller. On this panel, you can define which CC message will control which StarWaves parameter.

Just select the parameter on the *Param* dropdown menu which you wish to assign a controller. And then select the CC number on the *Midi CC* dropdown menu, which will control this parameter. And click on the **Store** button; this will store this assignment.



You will see the related destinations below the helper arrows for the parameter and the midi controller assignment. Each time you select an assigned parameter or a CC number from its menu, the arrows will remind you the relevant destination if they have a previous assignment.

If you would like to cancel a parameter assignment, just select it on the list and click on the **Erase** button, which will free the assignment for this parameter.

The slider next to the Midi CC dropdown menu will let you specify the depth of the midi modulation on that parameter. A full scale control would be the slider value 1.

There are 162 parameters which you can address with Midi CC controller. A certain Midi CC number can be assigned up to 4 different parameters at the same time. From synthesis parameters to camera movement, this has a great potential of live control.

The CC 0-127 value range is automatically normalized according each parameter assignment.

After you have finished your midi settings , you can use the **EXP** button to save it as a preset into the app Midi Settings folder by using the File browser. And you can load it back later with the **IMP** button.

If you want to save your settings on top of the last loaded preset, you can use the **Save in Place** button. You can select your connected midi controller device by clicking on the **Midi IN** button.

## OSC (OPEN SOUND CONTROL)

StarWaves will accept OSC messages from external applications and let you address its parameters with OSC messages.

The local port number for StarWaves is **8888**.

At the bottom of the Midi / OSC display you will be shown automatically the relevant OSC message format to use for each selected parameter on the dropdown menu.

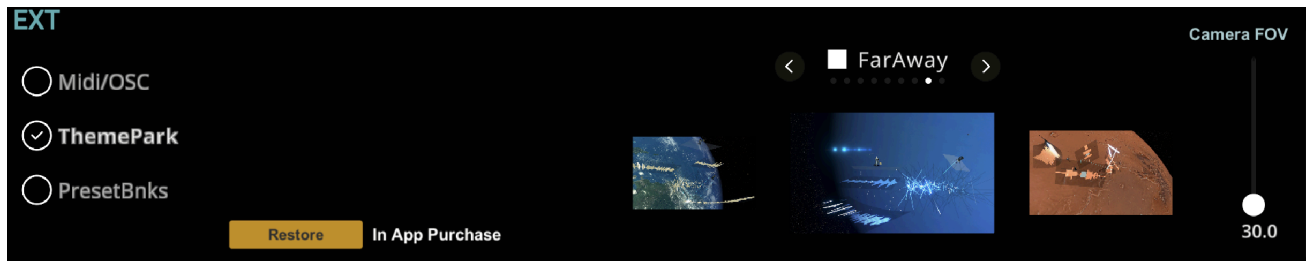
For instance, the OSC message **/starwaves/EM1duration 0.5** sent to StarWaves on this port number will set the Emitter1 particle duration to 0.5 seconds. Always use floating number format when sending the values.

## THE THEME PARK PANEL

The Theme park add-on panel will reveal itself automatically after the In App purchase process and restarting the app.

StarWaves comes by default with 2 scene themes : default theme and the milky way theme which put your scene in space full of stars.

With the Theme park add-on, 7 additional meticulously crafted theme designs will be added. Also an additional preset bank called *StarThemes* with 13 presets will be accompanied.



The Theme park panel controls are straightforward. You can choose your preferred theme for your preset ( this choice will be saved within the preset ).

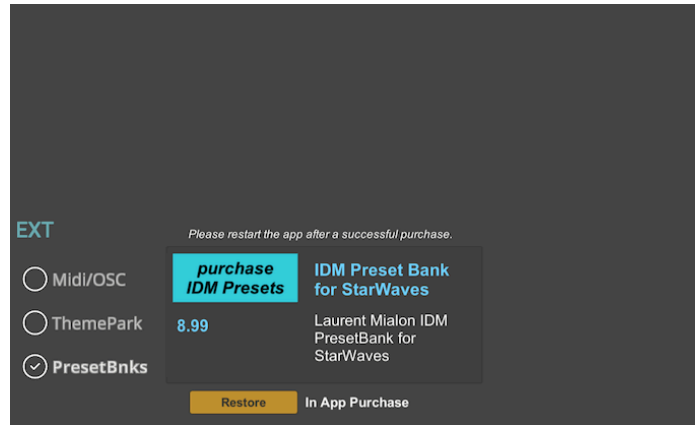
Also there is a Field of Vision camera control slider ( **Camera FOV** ) to give you more background depth control of your theme.



## ABOUT THE PRESETS OF STARWAVES

Here we present some comments of Laurent Mialon about his preset creations on StarWaves.

**Attention :** *The recent update of StarWaves includes also the IDM in-app-purchase preset bank of Laurent Mialon.*



## FACTORY BANK

### Blink\_If

Two rather rhythmical samples here have been loaded using the **Att Segment** feature, which divides the platforms according to their transients (vs. equally). Since they were not classic drum loops, during their making of, a transient enhancer has been used (many DAWs have such a tool, using a compressor with a slow attack can also help). A noise gate also can be useful, to further define the sample parts to be isolated by Starwaves.

When switching back and forth from the first to the second state, you can hear some drone produced by the second emitter synth engine. Its beam, unlike the other emitter's, is not moving, which contributes to giving a steady drone-ish sound. It is only audible during the state 1 to state 2 transition, as its volume is turned down for the first snapshot (yet you can see the trapezoidal shape made by the reflecting beam).

Even with non rotating emitters, the combination of a randomized area for the EM1 beam with some oddly placed platforms can produce some generative soundscape, with endless changes.

From state 3, the emitters are rotating, unveiling sounds that were not audible until then.

### Cinnabar

This preset was inspired by the sulfide mineral which is the most common source ore for elemental mercury.

By morphing between states which have different delay rates, these being often modulated by the **Yaxis** and the **Angle** settings, it's possible to achieve some sounds that feel at the same time metallic and liquid. It can be interesting to try out long sequences of snapshots with short morphing times to come up with organic structures.

## Growl

As for all the other presets, it's worth exploring the states one by one, as well as playing them sequentially (by pressing **Play** and even **Loop** - since both emitters will randomly scan the platforms, the probability to have the same repeating sequence is close to null).

One little trick here to create pseudo exponential rhythms was to have a beam hitting the same segment, with short duration sounds on one state, and increase that value for a nearby snapshot.

## Her Majesty's Sheep

A minimal preset, which is better played one state after the other, with manual changing of the state (**shift + state no**, if you are in full screen mode, double-click anywhere on the scene screen for this). By reversing the platforms, tilting or rotating them alongside their x axis, modifying the pitch shift amount between states by meaningful intervals, it is possible to get different atmospheres, which played slowly one after the other will tell a full comprehensive story. It's good practise by the way to load samples that are within the same scale, though experimenting is totally worth it too.

### In Pace

This preset is also better played slowly, one snapshot (state) after the other.

Let's focus on the first state to showcase a little trick. The camera on that snapshot is focused on the only audible events (others are coming in the other states, and are visually shown). When the beam touches platform 1 for the second time (after it's been reflected by P2 on top), it will touch the platform at a different (non perpendicular) angle than initially. With the **Angle** to Frequency slider pushed to the max, with a pretty high resonance, you can get some nice overtones - which are also randomized, as the beam will hit the platform surface on a random area (within the limits defined by emitter's **X dim**, in conjunction with the effective length of the platform, defined by the platform's X dim). It's thus possible, with a minimal set-up to make never-ending melodies.

## Iskander

For the first two states, the emitters are static, sending their particles on a chosen segment. It can be a visually rewarding experience to then play with the particles view, with some pseudo stroboscopic effects.

You can also note, that for the second preset, the original sound of P1 has been enriched with the platform's reverb, harmonically **Pitch** shifted. It's worth trying out different FFT sizes every time you use the pitch shifter.

From the third state, the emitters are rotating. In state 4, some particles are bouncing whilst the slowest rate emitter's aren't : the same sounds are being scanned in parallel with a slight delay related to the emitters relative position to each other's, but these synched emitters are doing way more than a simple echo, it's in turn like layering one sound with a totally different

texture - which happens nevertheless to be directly related to the platform's content, initially unveiled by the first emitter to hit it.

## Oxycomptine

A rhythmic preset, which basically consists of three platforms hit by the same emitter, whilst the second emitter is playing the background pad.

The colour of this last sonic event is changing from one preset to the other, thanks to notably some **Pitch** shifter / **FFTsize** adjustments. Also, to give the impression the pad is not static, the particles hit the platform in different spots on both axis, the **Yaxis** being a modulator routed to the platform's LPF (alongside the angle).

As far as the other discrete events (vs. the background pad) are concerned, the variety between states is a consequence of the platforms' position, enabling the particles to bounce on specific areas of them, at specific times - yet with a randomization factor, related to the area chosen for the particles emission, alongside with the degree of randomness per axis (**Mov** slider).

## Station F2

Only two emitters and two platforms for this preset.

The 3 beams of EM2 are scanning the second platform, where a sound made of the 4 microtonal (Bohlen-Pierce scale) distinct tones of plucked synth has been loaded. Sometimes you will hear a 2-notes loops, sometimes a 3-notes one, and eventually the full loop will play.

In the background a highly modulated sample made on Novation Bass Station 2 is being scanned slowly, for some classic granularization textures. The richer the texture of the loaded sample, the more variety you will eventually get once the platform is played within StarWaves.

When they are meant to overlap it can be recommended to load sounds with a different frequency spectra on different platforms, so that their frequencies stay audible.

Please note, notably on State 3, the comb effects on the Bass Station sample, here achieved with a short delay time on the EM1 synth engine, the modulation being provided by both the **Angle** and **Yaxis** position.

## STAR BANK 1

### AlaFX

This is a variation of the preset named Oxycomptine, present on the Factory Bank

### Being Palmer Eldritch

Here, 3 platforms have sounds with defined transients, hence the choice to use the **Att Segment** dividing mode. They are all together triggered by the same slow rate Emitter, whilst the tonal part is provided by a Tibetan bell loaded on P4.

Things get more complex from state 4. Putting the platforms closer to each other's and using the **Scatter**, **RotAnim** and **Parabolic** function establishes a topology that allows the formation of reticular sounding patterns that always change. By adjusting the width of **X dim**, we can effectively control the amount of randomness.

Be sure to try out slow and fast morphing times when you play the sequences.

### CDG Nord

A simple setting that works well by default is to have the platforms in vis-à-vis, with rotating emitters reading their content - one emitter for each pair, emitters being 90° apart.

Between states 2 and 3, the division of P3 and P4 has been increased, transforming a discrete arpeggio into a more continuous sound (still with transients actually, but closer to what we could refer as a single iterative sonic object). Using **RotAnim** alongside the assignation of **Yaxis** to the sound engine's delay time combined with the modulation of the LPF cutoff by the **Angle** of collision transforms what could be a static iterative sound object into a more organic iridescent, swell.

### Cinq Cents

This preset was conceived thinking of what StarWaves could also be about (the fascinating relationship between the void and silence, space and sound), with arguably Pascal's most famous quote (« le silence éternel de ces espaces infinis m'effraie ») loaded on P1.

Let's jump to state 5, and focus on that classic psychedelic effect on the voice : as the synth engine delay time is increasing the perceived pitch of the sample is, as both the platform **Pitch** and EM1 **Delay time** have a positive modulation originating from the same source, **Yaxis**.

Let's go back to snapshot 4, and consider one of the advantages of using a twisted platform. Besides some possible variations in color (in the event the position on the Y axis and the angle of the collision are used as modulation sources), we can use the odd shape in conjunction

with a scanning beam to have irregular reflections on further platforms. Here, what's being reflected on P3 reaches P4. Try changing P4's **Parabolic**, **RotAnim** and **X Dim** parameters to observe how the beams originating with EM2 end up on either platform.

P1 is being initially scanned by the other emitter; the occasional playback that is a consequence of the multiple reflections of the EM2 beams contributes to the repetitive hypnotic effect.

## Crevasses

Let's focus on the action of EM1 (rotating alongside the vertical axis) in relation to platforms 1 and 2 (the horizontal ones).

Recall state 1. Both platforms have been divided in seven segments. Rare particles (it's a slow, irregular rate - note the high value of **S rand**) are triggering specific piano chords.

Now recall state 2. The platforms' segmentation has been pushed to the max, transforming the loud piano chords into accelerating or decelerating sequences of the same chords - the change in speed being controlled by the curvature of the platforms and distance related to the emitter. Also, by reversing the platform orientation, the beam will pick various amplitudes of the decaying chords, thus creating, depending on this orientation, either a crescendo or a decrescendo.

## Dub Huit

A simple preset, where a rhythmic loop is being read by the first emitter, whilst the second is responsible for playing a background pad. Now and then EM1 also triggers the playback of the other platforms' blocks.

This is particularly noticeable in state 4, where the twisted shape of platform 2 allows the occasional triggering of some of P4' blocks, creating some harmonically rich background.

## Kinzhal

A melancholic ambient preset, with little changes between the states. Please note though, the difference in tone that is a direct consequence of the FFTsize adjustments between states.

## Mandragora Blooming

One emitter is sending beams to 2 platforms, that have been segmented with **Att Segment**, given their nature, and more important, the sonic intention (remember that you can always edit the sounds to be loaded so that StarWaves reads a specific number of blocks, either accentuating transients, either making some perfect divisions).

## Morgen

A few things can be spotted here.

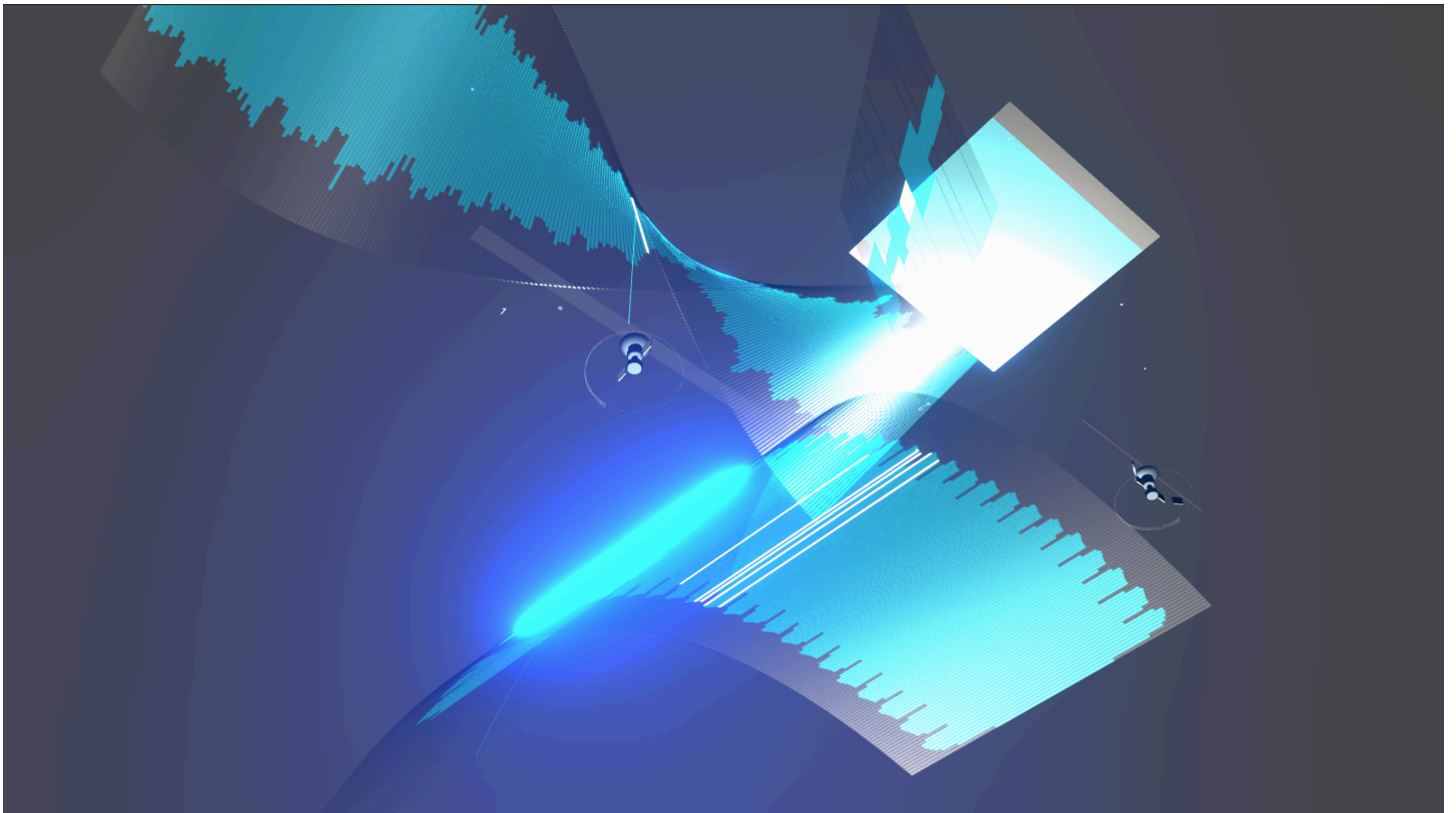
The State 1 to 2 transition : for EM1, check the sliders of **EM dur**, **EM rat**, **Em Spd**, **Y dim**. For the platforms' FX, check the following **PL1 FX** parameters : dry and wet gains, pitch shifter, FFT size. By changing multiple parameters at both the collided platform and the emitter synth

engine levels, you can use StarWaves to create the most unusual transitions and with the added advantage of having a powerful visual feedback.

On State 5, the platform blocks have been fractured and made fitting the same small space, with the first emitter being in the core of that topological chaos. Since the emitter is rotating, with big values for both **X dim** and **Y dim**, you have now a totally self-generative abstract mini piece, which sounds totally different from the straight loop you could hear in states 2,3,4.

# STARWAVES

Finally, thank you for using StarWaves. Independent of your profession, and what you are doing, we believe that it will spark some creativity for you and deliver valuable experience for the time you spend on it.



[WWW.SONICPLANET.COM](http://WWW.SONICPLANET.COM)