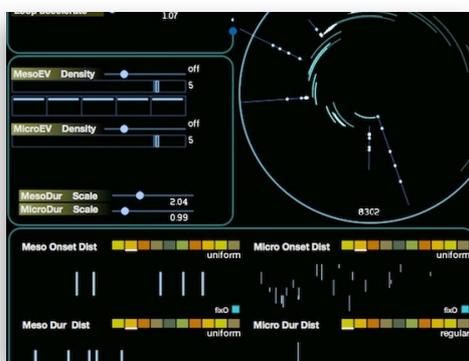




Cosmosf M31 and MBot is conceived, designed & programmed by Sinan Bökesoy.

in VST/AU programming platform: *JUCE / C++*



all rights reserved ©Sinan Bökesoy, sonicLAB™

www.sonic-lab.com

installation of Cosmosf M31 :

Cosmosf M31 is a stochastic multi timescale midi effect plugin. And it comes with Cosmosf MBot. They generate MIDI event data to be used with other synth plugins.

Please unzip the downloaded files and keep a copy of it but erase any old versions existing in your HD.

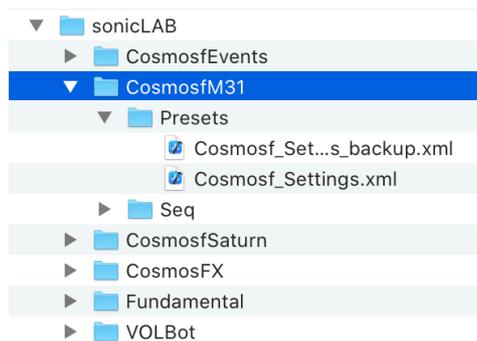
(For the OSX version)

- **CosmosfM31.component** and **CosmosfMBot.component** files : ideally should be copied to your system / **Library/Audio/Plug-Ins/Components** directory in your system harddisk. For example */Volumes/OSX-Main/Library/Audio/Plug-Ins/*
- **CosmosfM31.vst** and file **CosmosfMBot.vst**: ideally should be copied to **/Library/Audio/Plug-Ins/VST** directory in your harddisk. For example */Users/sinanbokesoy/Library/Audio/Plug-Ins/VST*

(For the Windows version)

- Please put the unzipped folder **CosmosfM31_MBot** in **Program Files / Common Files / VST2** directory of your system harddisk. Don't take the files outside of this directory.

On OSX : When you launch *Cosmosf M31* for the first time, it creates a *sonicLAB/CosmosfM31* directory in your *Users/Shared* directory on your harddrive. There it automatically copies all the files it needs to communicate.



On Windows : When you launch *Cosmosf M31* for the first time, it creates a *sonicLAB/CosmosfM31* directory in your *Documents* directory on your harddrive. There it automatically copies all the files it needs to communicate.

To run the *Cosmosf M31* plugin , you have to trigger it with a Midi note event on your DAW. (only one note). This will be also the basis key for the generated events.

Cosmosf MBot can run only with M31 together, when they are both connected. M31 becomes the server and the multiple MBot's can be its satellite clients.

some important facts :

MIDI resolution: For a feasible MIDI resolution on *Cosmosf M31*, don't select a large audio buffersize (max 512) nor a small buffersize (such as 64).

Very fast events, with high density (duration of few miliseconds) might be omitted, or will not receive any response from the synth plug in chain.

*DAW's apply different tricks to process tracks simultaneously. On *Cosmosf M31*, we have tried to obtain best stability and synchronous behavior among multiple tracks to avoid problematic cases when you run the same MIDI effect on multiple tracks.*

*Time stamping of dense MIDI events (on a complex app such as *Cosmosf*) means taking careful measures. At the moment *Cosmosf* calculates all the events at the start of each macro event, and most of the parameter changes on the UI will be applied on the next macro event cycle, and not the moment of your gesture.*

MIDI channel output: *Cosmosf M31* will use the default midi channel 1 to send its generated events. Alternately you can use the multichannel output mode where each micro event will use midi channels between 2 and 14. You are welcome to use MPE synth plugins, as *Cosmosf M31* offers the compatible multichannel midi output mode, hence starting with MIDI channel 2 and going upwards for each micro event.

Information contained in this manual is subject to change without notice and does not represent a commitment on the part of sonicLAB. The software described in this manual is provided under the terms of a license agreement or non-disclosure agreement. The software license agreement specifies the terms and conditions for its lawful use. No part of this manual may be reproduced or transmitted in any form or by any purpose other than purchaser's personal use, without the express written permission of Sinan Bokesoy.

introducing the key shortcuts

You should grasp some essential information about the key shortcuts of *Cosmosf*, since they don't have a button/slider equivalent on the graphical user interface. Hence;

- < **Any note on your midi keyboard** > is equivalent procedure as the above case. *Cosmosf* M31 starts with note-on and ends with note-off.
- < **z** > Resizes the UI windows size, pressing it switches between sizes of factor 1, 1.2 and 1.4.
- < **p** > When pressed *Cosmosf* receives the host tempo value from the DAW and applies this to the MacroCellLength. Likewise the cycle lengths stay in sync with the host tempo changes. Pressing the key again deactivates it.
- < **r** > Starts recording on the Object engine.
- < **q** > Turns on the Object engine playback.
- < **e** > This key cleans the Object Sequencer tracks all at once. While you are recording and preparing a *Cosmosf* object, this key will simply clean your object.

how to study *Cosmosf* M31 :

Cosmosf M31 is one of most advanced event generators and also a complex one, a different paradigm in comparison to other relevant tools. Therefore it needs time to study, experience, but it will be a very rewarding effort in return.

It is essential to go through the manual and try out the existing features. There are of course topics which are not handled here in depth detail, such as stochastic functions. You can find many resources on the internet on various levels.

Every step of the *Cosmosf* structure is driven by the language of mathematics mapped to sound synthesis. There is logic, transparent interaction and an enormous amount of potential to sonic emergence.

presets of *Cosmosf* M31 and example projects :

Cosmosf M31 presets are kept simple and they try to feature some potential of the software. With simple gestures on each preset *Cosmosf* may become the computing engine preparing the impressive results which can't be done by hand and by other software tools.

This is a stochastic synthesiser possessing random iterations on multiple time scales, which means things might probably not the same each time you play the presets and expect to behave the same way. As Heraclitus stated "You cannot enter the same river twice" !.

Cosmosf comes with one bank of 16 presets. We don't believe that presenting 100's presets are pushing the creativity and also that creates a tendency in shuffling through the presets rather than exploring the possibilities. In sonicLAB products, it is the tiny bit of alterations and gestures which open up the most interesting results.

We have prepared example projects for new users of *Cosmosf* M31 and MBot. These Logic and Live projects will help you understand quickly how to setup the M31 and MBot on these DAW's.

Please check them at this folder; you will find also a *readme* file describing each project.

https://www.sonic-lab.com/Cosmosf_files/M31_projects/

There is still so much to put in *Cosmosf*, and they will be showing up as the planned updates of the application. For instance in 2012, the V2 of the standalone version, has offered a genuine feature, the ‘parallel universe’, which replicates a *Cosmos cycle* with its event generation mechanism but with fully independent synthesis engine from the latter. V2.2 introduced a rich and intuitive morphing mechanism all with specially designed visualization support.

Finally the VST/AU version of *Cosmosf* has arrived, which is probably on of the rare examples that a research product with such a rich outcome enters to the domain of professional music production platforms. And with the release of *Cosmosf Plugin V2*, the software gains enormous potential as you can independently define the synthesis structure of each micro and meso event and operate the powerful modulation tools on them. It is a wormhole to complex sonic structures.

Cosmosf vSaturn includes the “sieves” engine to construct all the musical scales to be used on various levels of the synthesis engine. It also opens a new world to Midi performance possibilities.

Cosmosf vSaturn supports surround sound audio with independent movement control of its meso-events in surround space by 2D/3D manipulation possibilities. The latest version *Cosmosf Saturn v2* is a high precision digital audio lab with impeccable sonic quality including high quality filters.

Cosmosf Saturn v3 is the excellence reached after years of development. It includes dual macro events with overlapping ability, increased meso and micro event density and highest precision and stability for event generation mechanism.

Cosmosf v4 and *Cosmosf FX v4* is a total rewrite of the existing code, an enormous amount work to implement new features, extreme stability to carry on for the next developments. One of the most advanced sound synthesis tool of our times.

Cosmosf Saturn5 and *Cosmosf FX5* introduces a truly modern and professional studio tools serving as an instrument, algorithmic soundscape renderer and a modern surround sound production tool for serious sound design tasks.

Cosmosf Saturn6 introduces the “Object” a 2150track parameter sequencer which records your gestures in various ways and plays them back in various configurations.

Cosmosf Saturn7 introduces the “Grid” system, which lets you define and edit the onset positions of the meso and micro events. It also offers degrees of stochastic transitions between fully deterministic and fully stochastic.

Cosmosf Saturn8 versions are refined and improved versions with various fixes and boosts.

And finally *Cosmosf M31* arrives with the sonicLAB Bot technology embedded. It s a midi event generator, horse power presenting its data to other synthesis plugins.

wish you a creative day,

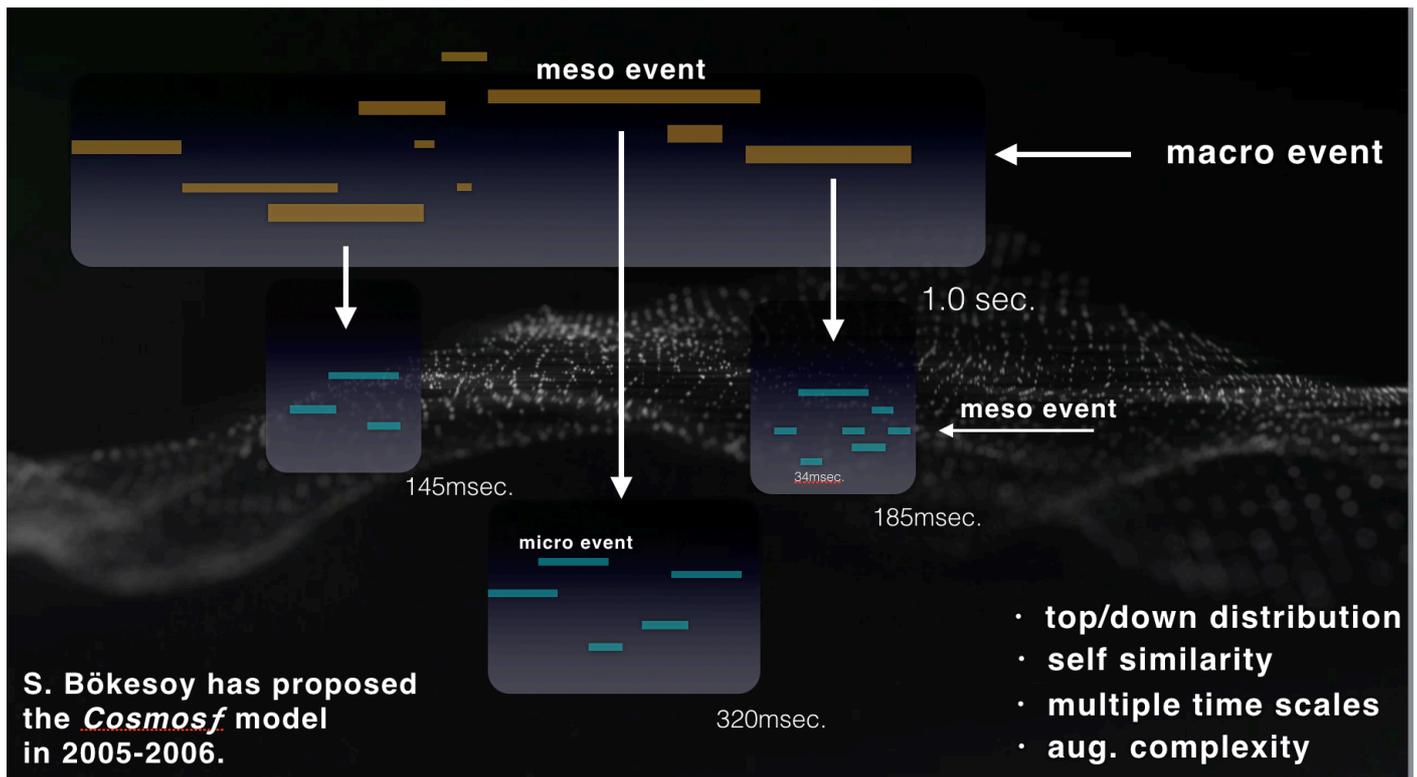
Dr. Sinan Bökesoy., January 2021

chapters:

The stochastic event distribution process	8
Structure and the controls of the main screen	11
The modulation sources of <i>Cosmosf</i>	17
Note patterns and transition to controlled randomness	20
introducing the “Object” engine	21
Automation in DAW with <i>Cosmosf</i>	25
Preset Import / Export	26
<i>Cosmosf</i> M3I and MBot : a fleet of Bots	27

The stochastic event distribution process

COSMOSf STRUCTURE



The top-down event organization in Cosmosf:

Music is about events; sonic components are organised in time as they arrive to our ear through the vibrations in the air. A note drawn on a score, or a midi event on the editor of a sequencer or a phrase of recorded sound in a track of your DAW, they all exhibit an event with specific start and stop time and certain evolution during its duration.

Xenakis has introduced the use of statistical mathematics for the organisation of notes and sequences and he used this approach extensively in his works beginning in late 50's. He has used different types of stochastic functions to decide the distribution of musical components and configuration of the structure. His ingenious approach led to the roots of sonic granulation in computer music and some computer aided composition as named algorithmic music.

Cosmosf is an advanced stochastic event generator and organiser distributing sonic elements on multiple time scales along with complex modulation scheme. As you see on the figure above, we define the whole being of Cosmosf as a *macro sonic event*, having a certain duration titled as **MacroEV Length**.

Meso-events of certain density are distributed inside this time space with their **onset time** and **duration** parameter calculated by stochastic functions. Also the **meso event density** can be calculated with stochastic functions.

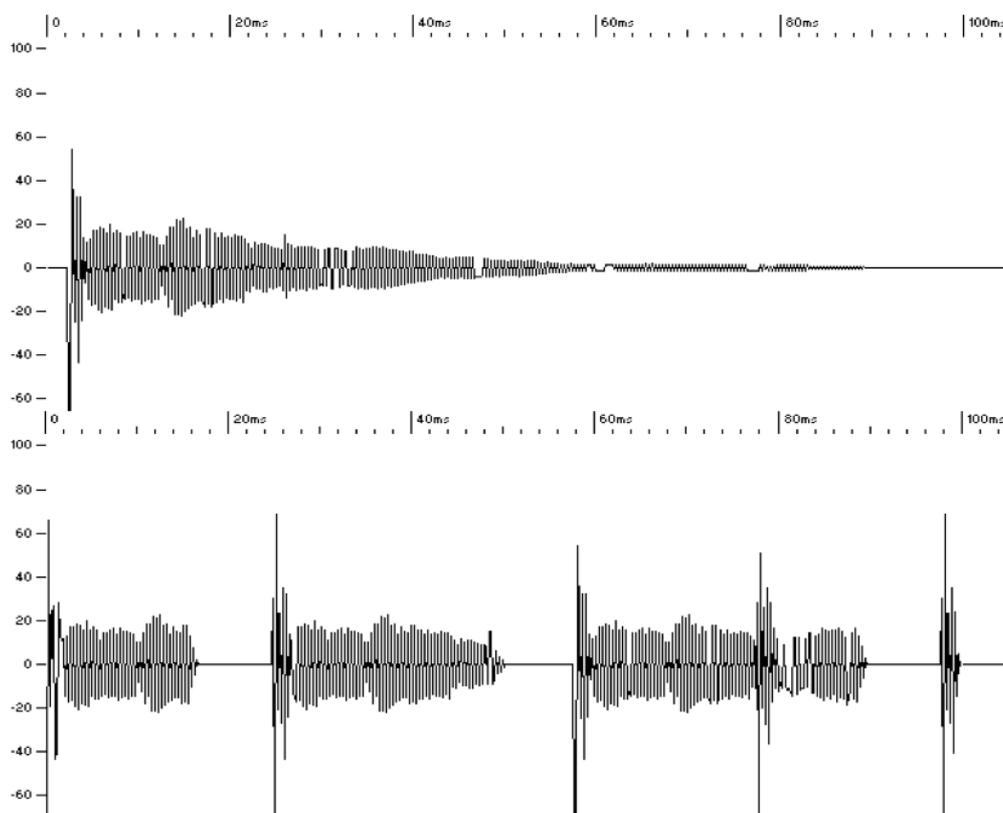
Each meso event defines a **micro-space** in the Cosmosf event distribution system. In each micro-space, events of certain density are distributed with stochastic functions. They are called **micro-events**. The micro-events are the sonic elements in Cosmosf; they will carry the waveform information assigned by the **Synthesis-Engine** to them. The micro-events or meso-events can overlap each other.

The event distribution process in *Cosmosf* follows the hierarchy explained above whenever there is a new macro event to be filled with meso events; and then instantly they will be filled with micro events. It does calculate the necessary onset time and duration values with the given density values and assigned stochastic functions.

This self similar distribution structure augments the complexity of the distribution process proposed by Xenakis, while keeping the system integrity. Like in the nature, the organisation happens on multiple scales; the local/macro density, the dynamic behavior and therefore the perceived complexity.

We have mentioned that the micro-events are carrying the sonic entity assigned by the Synthesis-Engine, but what does the meso-events do? Actually the micro-events accumulate each other and become a meso-event. In other words, a meso-event is itself a sonic-event consisting of micro-sonic events generated by *Cosmosf*. On the upper part of the figure below, you will see the waveform assigned to the micro-events. And the waveform below represents the meso-event waveform constructed with the accumulation of the micro events.

COSMOSf STRUCTURE



The meso-events accumulate in time to construct the macro-event, which is the output of *Cosmosf*. And here comes another unique feature of *Cosmosf*. The output can be fed back to the micro-level events. This opens the doors of self evolving emergent sonic character. By introducing a hierarchy of multiple time scales for the event distribution process; modulation sources constructed with stochastic - deterministic functions, it is possible to form sonic creations that cannot be handled by traditional synthesis methods. The user intervenes with the system in real-time by inputting a sound source at the micro level and interacting with the system by controlling the parameters for the sonic event generation on different time scales and parameters for different type of synthesis/modulation generators.

Cosmosf uses probability distribution functions, also named **stochastic functions** to decide the density, onset time, and duration parameters of each meso and micro event. For instance the following table is a list of the stochastic functions operating in *Cosmosf*;

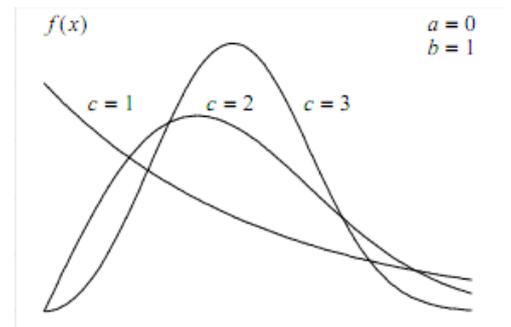
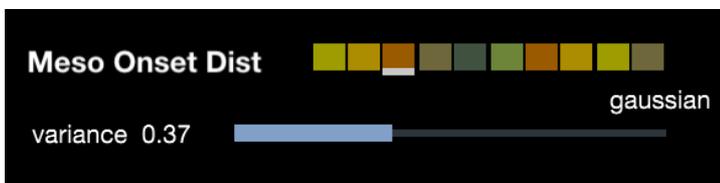
Density Distributions		meso-space density	micro-space density	
poisson distribution				
uniform distribution				
gaussian distribution				
<hr/>				
MacroEV Length				
uniform distribution				
gaussian distribution				
<hr/>				
Onset time & duration Distributions		meso events	micro events	
uniform distribution	cauchy distribution	exponential distribution	weibull distribution	triangular distribution
gaussian distribution	exponential distribution	arcsine distribution	lognormal distribution	

A probability distribution function shows distinctive behavior when distributing the values as its output. For example we want to distribute micro events in a micro-space. We know the micro-space length from the relevant meso-event duration. And now to obtain the duration of the micro-events with some density, we call a stochastic function, which will decide for these values.

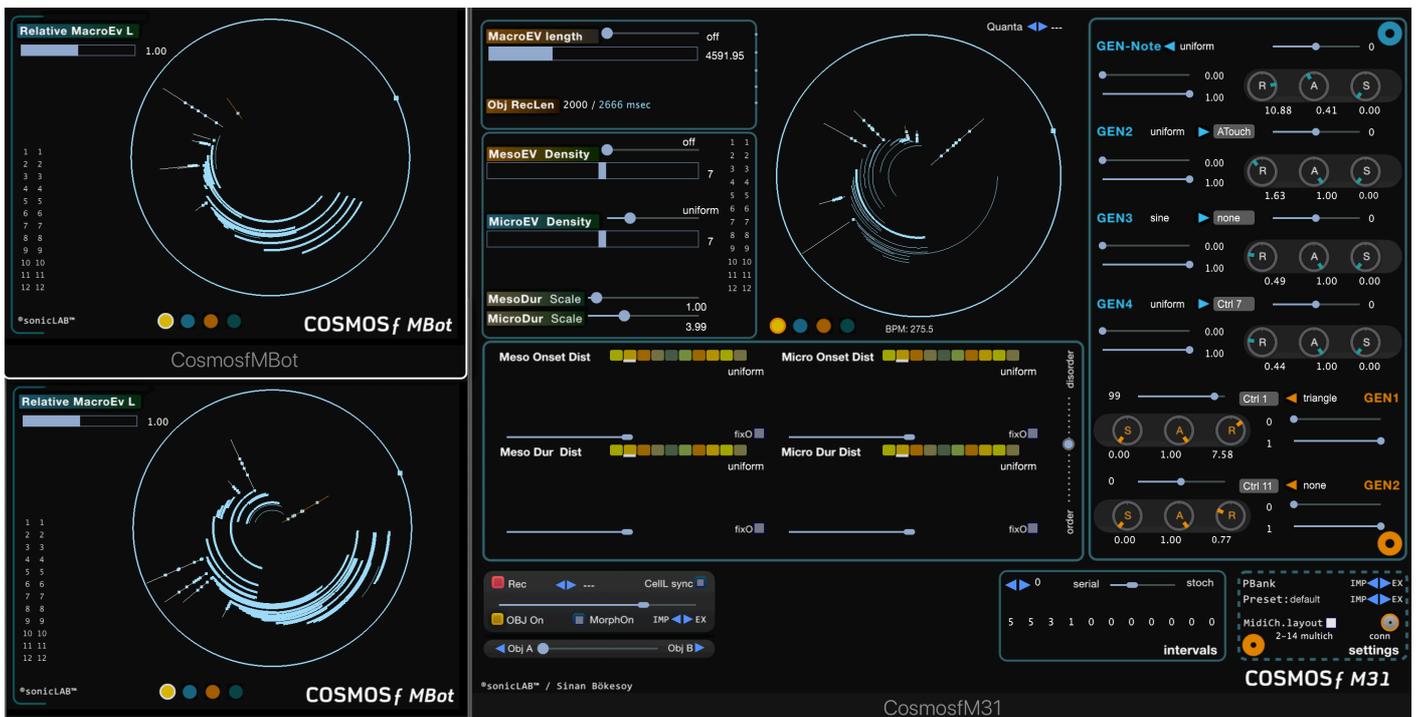
The duration distribution should be between 0 - 'micro space duration'. We call this function the same amount of times as the micro space density. Most of the stochastic functions have parameters, a value of certain range to change the behavior of the stochastic function.

The reason that *Cosmosf* has many types of stochastic functions is to offer to you different types of 'distribution shape', which gives a distinctive character to the process being used for.

On the figure below the gaussian distribution function has been assigned to the Meso-Onset distribution. The parameter is set to .37. The 'distribution shape' of the Weibull distribution is given on the right with three different parameter settings (1,2,3). If you are interested to know in detail about stochastic functions, this link is a recommended reading;



Structure and the controls of the main screen



Cosmosf M31 separates itself from other Cosmosf apps by not possessing any digital audio content on its micro events. Instead each micro event will be delivered as Midi note on/off event to the output of the plugin. And the modulation sources will be producing Midi CC messages.

The top-down event organization in Cosmosf is controlled within the panel on the top left side of the main screen.



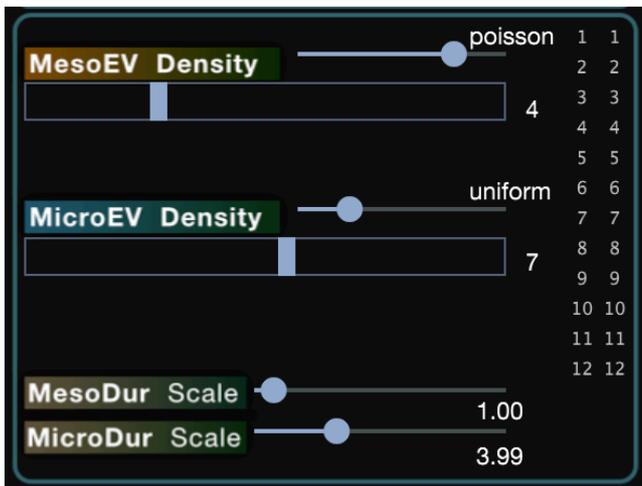
The macro event length can be set with the slider under the **MacroEV length** label. The value is set in milliseconds. You can also use stochastic functions to define the MacroEV length. With either **uniform** random or **gaussian** random distribution functions, one can calculate the MacroEV length after each full cycle of Cosmosf. The type of the stochastic function can be set with the point slider next to the MacroEV length label.

By pressing the “p” key you can sync the length of the MacroEV to the tempo of your DAW. Increasing the tempo will shorten the MacroEV length for example.

The Obj RecLen value fields serve for the record buffer length of the Object engine which will be discussed later.

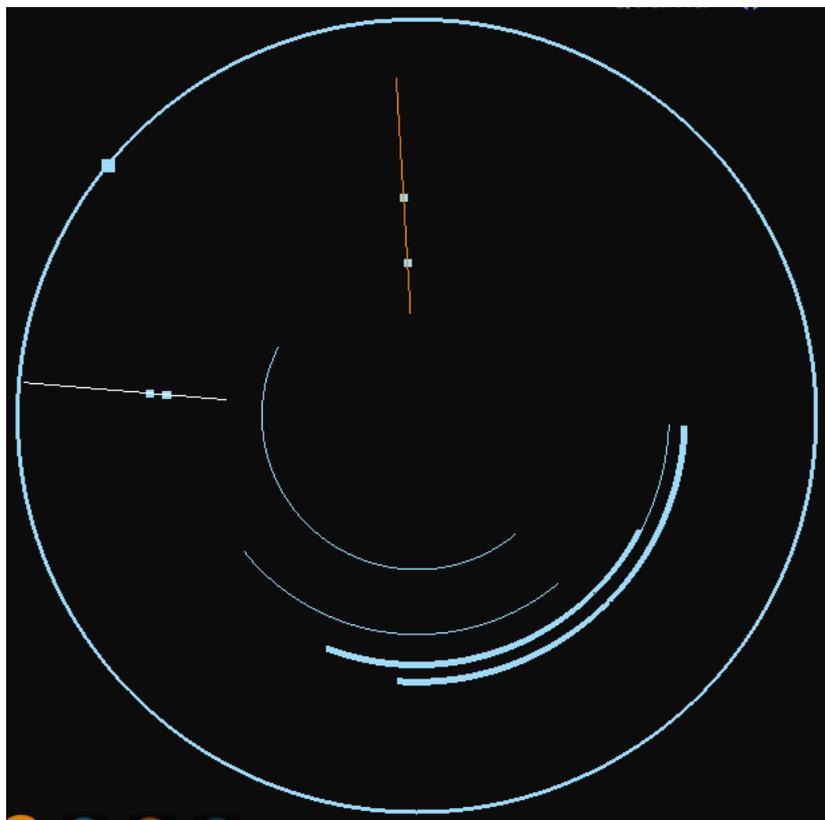
The main screen panel in the figure below has controls to define the event density distribution process for the meso and micro events. The meso event density in the meso space and the micro event density in the micro-space can have a value between 0 - 12

Let’s say you set 5 for the meso event density and 4 for the micro event density. The total amount of micro-events in the Cosmosf cycle is then $5 \cdot 4 = 20$. The density can be controlled with stochastic functions such as the **uniform**, **gaussian**, **poisson** and **binomial** distributions. The maximum micro-event density is $12 \cdot 12 = 244$, which will be distributed in a Cosmosf cycle with calculated onset and duration parameters.



With each new cycle a new **MesoEV density** will be assigned, and with each new meso-event onset, a new **MicroEV density** will be calculated. To set the maximum density amount for a distribution, use the sliders with the blue bar. *Cosmosf* shows the current calculated density with the unfilled blue bar on the same slider.

The meso and micro event durations can be scaled by the user, which means each meso/micro event duration value will be multiplied with the value set on the sliders **MesoDur Scale** and **MicroDur Scale**. Scaling the value of the event durations do affect the overlapping between them. These can be set individually for each micro and meso event too.



The *Cosmosf* cycle visualization gives you an instant look to how the event distribution process generates the meso and micro events in each cycle of *Cosmosf*. Around the circle you will see the small dot turning anti clockwise to show you the time. A full circle equals always the to cycle time as calculated by the MacroEV length and its distribution. The straight lines on the circle define the onset position of meso-events distributed in the macro-event. The small circles found on each meso-event line represent the micro-events. When their onset time arrives the corresponding arc around the center of the circle. the end of the arc is the end of the micro-end duration. The micro-events are positioned on the relevant meso-event line relatively to each other according to their onset time. The length of the meso-event line represents the meso-event duration. The arcs drawn on the screen represent each micro-event path with the absolute onset time as the start of the arc and the duration of the micro-event as the arc length. You can easily observe the overlapping events, their lifetime etc. information compact enough to grasp at one look.



Another feature is the possibility of **muting/unmuting** the meso events and the micro events with individual switches laid out on the event density control panel. The switches on the left column are dedicated to each meso event; when they are muted the button is in select state. The right column is dedicated to the switches muting the micro events. For instance if you click the 2nd switch, then every 2nd micro event in a micro-space is muted.

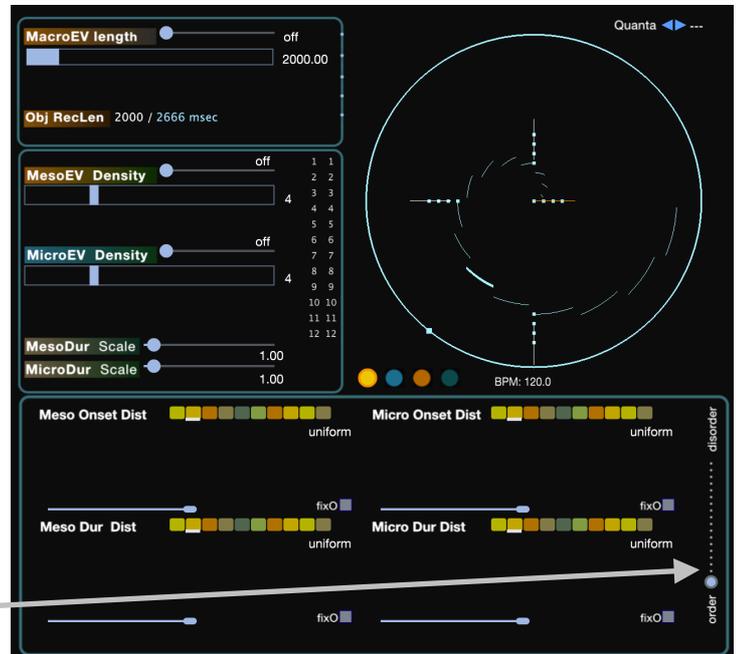
This feature is very handy in order to bring rhythmical silences in the flow of *Cosmosf* cycle, also to deconstruct / construct the macro object on the event level.

The bottom left part of the main screen is dedicated to the control set for the stochastic functions distributing the meso and micro event data. For instance each color button assigns a stochastic function for each distribution. You can vary their parameters with the slider showing up below the color bar area.

For instance the figure on the right shows an example case with 4 meso event and 4 micro events which seem to be uniformly distributed inside a 2000msec long macro space. Hence the result is a spiral, from the center swapn to outer limits, like a spiral galaxy.

On *Cosmosf* M31 there is the possibility to make transtions between fully deterministic and fully stochastic behavior.

The vertical slider order-disorder is responsible for this action. When this slider is at bottom, the event distributions (onset / duration) will be uniform as on the right. When it is at the top, the full stochastic values out of meso and micro event onset/duration distributions are being placed in.



The same setup but now the order-disorder slider is near mid position. The events are distributed all over in the macro space. What is happening ?

As you can see, the Meso and Micro Onset and Duration distributions are set to uniform random distribution. They were not being effective before, because the order-disorder slider was at the bottom, in total order state.



EVENT DISTRIBUTION

A range of statistical distribution functions are available for Meso and Micro event onset and duration values. And each of them have parameter sliders which will effect their behavior. We can only say, that you should try all of them and see / hear the results to understand their behavior.

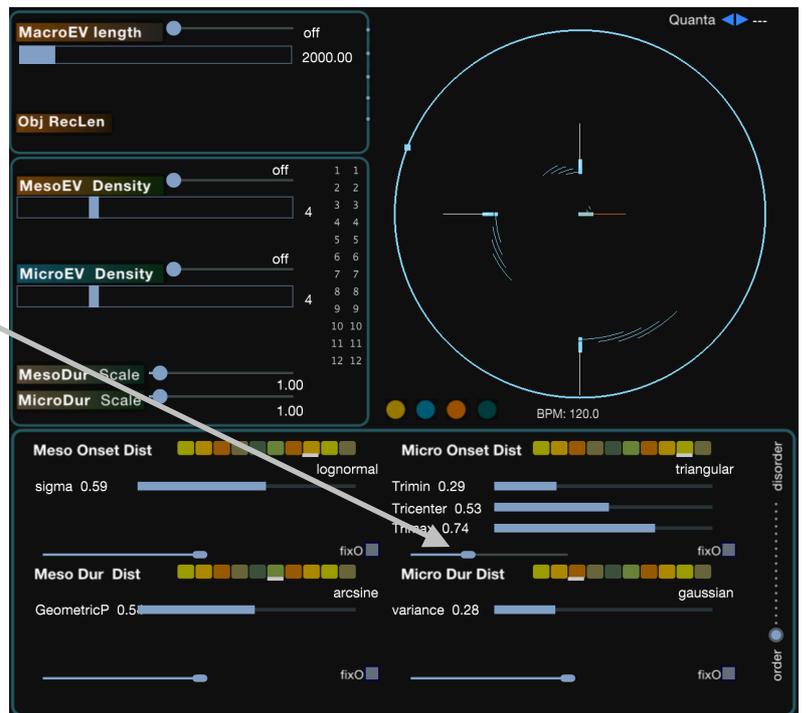


Now let's explain the gravity sliders for the meso and micro event spatial distributions.

When these sliders are moved to the left, the corresponding event values are pushed to “zero” values. For instance, on the right, we have moved the Meso Onset Dist. gravity slider to the left and all the meso events are squeezed towards the center of the macro space.



This example on the right shows the effect of the Micro Onset Dist. gravity slider. Now all the micro events are pushed to the beginning of their meso event space.



These small buttons labeled as “fixO” fix the distribution results for the next macro event cycles. The last distribution values will repeat and therefore you will create a unique looping pattern.

The screenshot shows a software interface with several control panels. At the top left, there are sliders for 'MacroEV length' (set to 2000.00), 'MesoEV Density' (set to 4), and 'MicroEV Density' (set to 4). Below these are 'MesoDur Scale' and 'MicroDur Scale', both set to 1.00. On the right side, there is a circular visualization with a BPM of 120.0. The bottom section contains two columns of distribution controls. The left column has 'Meso Onset Dist' (lognormal, sigma 0.59) and 'Meso Dur Dist' (geometricP 0.5). The right column has 'Micro Onset Dist' (triangular, Trimin 0.29, Tricenter 0.53, Trimax 0.74) and 'Micro Dur Dist' (gaussian, variance 0.28). Each of these distribution controls has a 'fixO' button. Two arrows from the text on the left point to the 'fixO' buttons for 'Meso Onset Dist' and 'Meso Dur Dist'. On the far right, there is a vertical 'order' control with a 'disorder' indicator.

Modulation Sources and Controls in Cosmosf M31

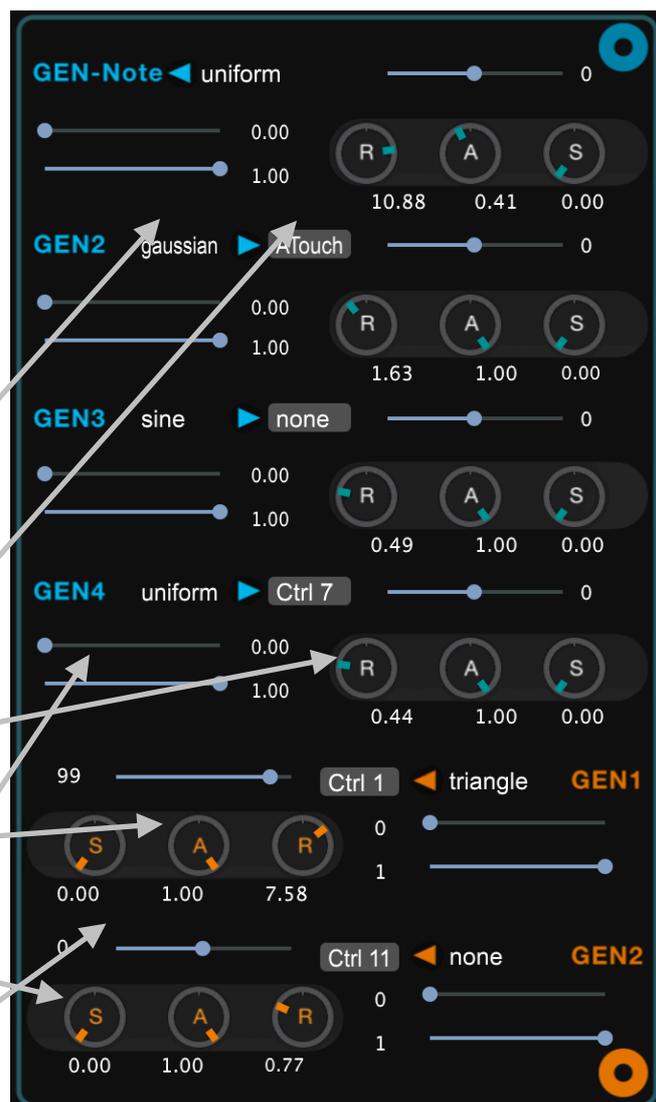
GENs

Unlike *Cosmosf Saturn* and *Cosmosf FX*, *Cosmosf M31* uses GEN's which are reminiscent from the sonicLAB BOT plugins and sonicLAB Fundamental. They are stochastic low frequency oscillators which offer a wide range of statistical distribution functions to generate their output.

Cosmosf M31 runs 4 individuals GEN's for each micro event and 2 individuals GEN's for each meso event.

For each GEN you can select the following;

- A stochastic distribution function but also standard waveforms
- The destination MIDI CC's, midi velocity and note value (only for GEN1 which is hardwired to note values)
- The generation rate value of the GEN.
- The modulation amplitude of the GEN.
- The degree of output smoothing of the GEN.
- The stochastic distribution parameters of the GEN.
- The offset value between -127/127 added to the GEN.



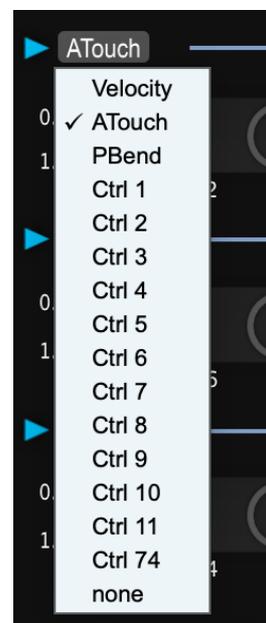
You can select a “none” value for both the stochastic distribution function and also the GEN destination. This will make the GEN sort of inactive.

Midi Note Value and Midi Velocity are discrete values, hence there is only one value of them assigned to the micro event during its life span.

However Midi CC's are continuous controller midi messages. On *Cosmosf M31*, the resolution of CC's is a buffersize worth of milliseconds.

As said, the micro event GEN1 is hardwired to Note Values, but the other ones can be assigned to the Midi messages of choice on the right. *Cosmosf* will automatically normalise the GEN output accordingly.

The only difference of the meso event GEN's is that the Midi Note Value and Midi Velocity destinations are not available to them.

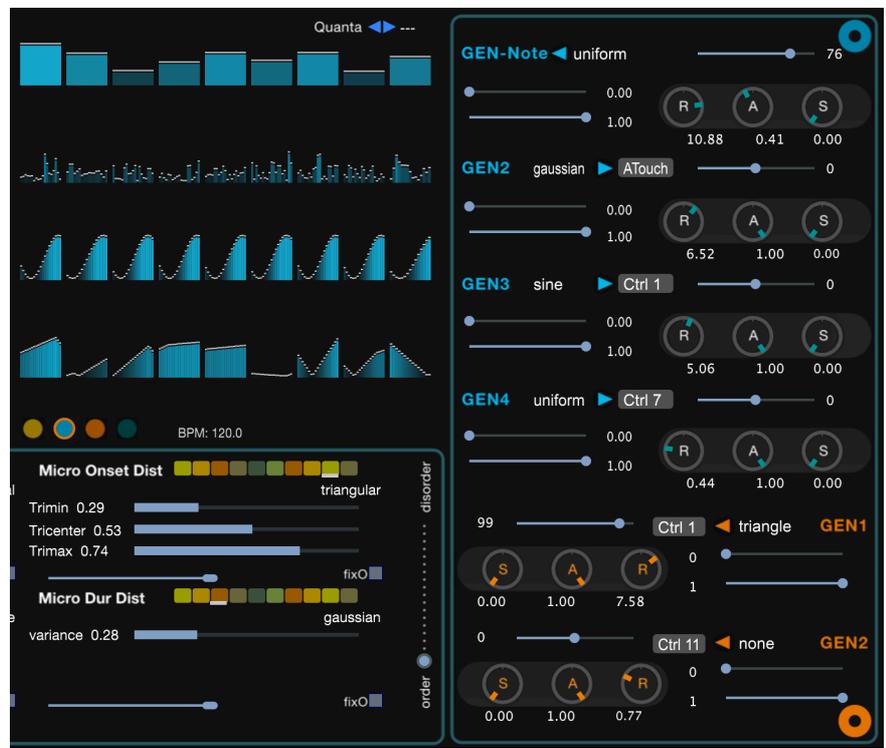


The visualisation dedicated for both micro and meso GEN's help to watch the live on going modulation scene. For instance on the right, the top row corresponds to the stochastic note value distribution of GEN1. You can easily see the discrete values assigned to each micro event.

The second row shows the fast gaussian distribution create a modulation with the AfterTouch midi messages. Please note the ongoing modulation during the life span of each micro event

The third row (GEN3) creates a sine wave like CC1 modulation.

And the fourth row (GEN4) creates a slower uniform random distribution for CC7 messages.



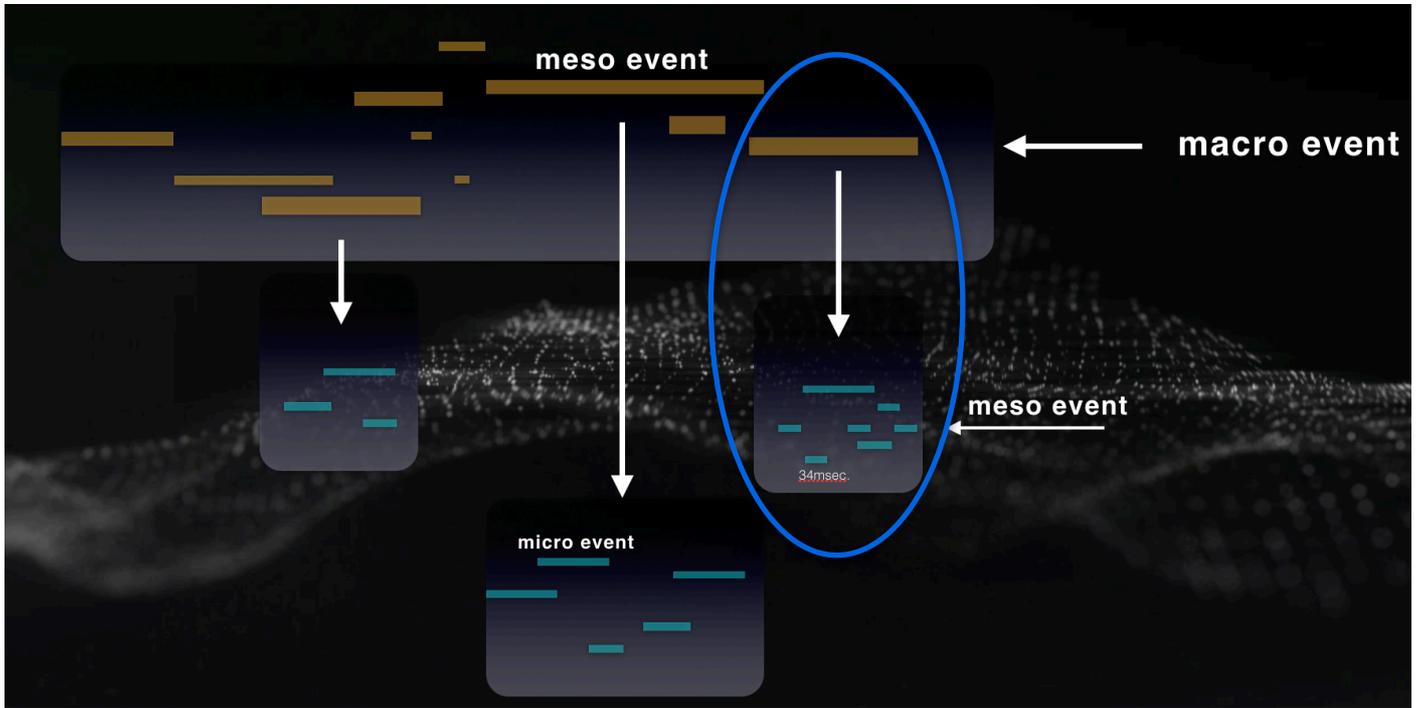
For the meso event GEN's the visualisation is similar. On the right, we see the effect of the chauchy distribution with its sparse peaks assigned to CC1 midi messages as well.

The second row shows the discrete Poisson distribution applied on CC11 midi messages. Discrete distributions create only one value for each event.



Now the killer question : **What does it mean to have meso scale and micro scale Midi controller messages together ?**

The answer is that you can generate Midi CC modulations both on the life span of meso events and the micro events. A meso GEN will create a modulation curve from its start to the end, and all the micro events during the time frame will be receiving the generated data and assign it to the relevant destination Midi CC which is the meso GEN destination.

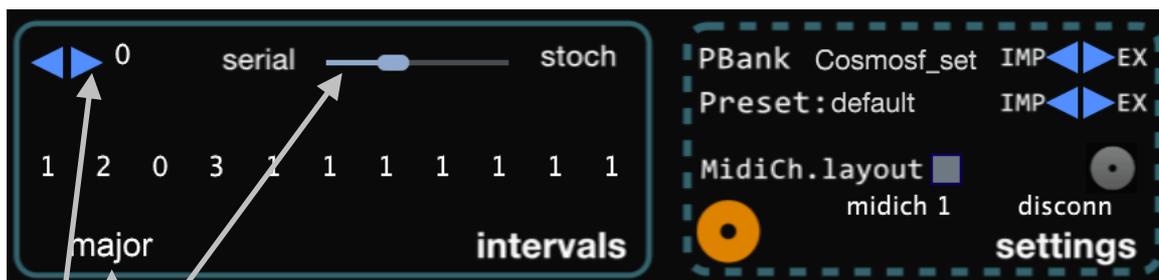


When both a meso GEN and a micro GEN has the same destination, then their modulation data will be added. This generates a more complex curve on multiple timescales. Likewise this sum will be output to the corresponding destination from the micro events.

Note patterns and transition to controlled randomness

On *Cosmosf* M31, you can define note value relations between micro events, a sort of serialism which works as following;

The note interval value can be typed directly on the intervals panel shown below. Hence given that the first micro event has C4 for example (input midi key note to the plugin) then according the values below the next micro events would be F4, A#4, C#5, D5. The zero interval value means that the last note will repeat.

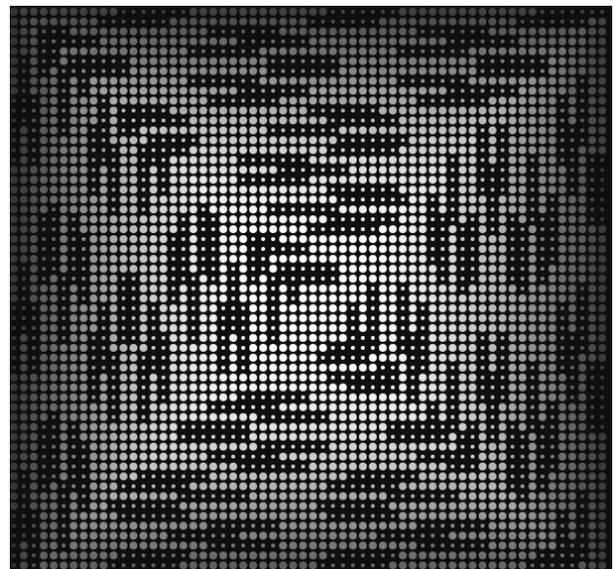


An important slider, the “ serial - stoch “ will let you impose smooth transitions from the deterministic pattern to stochastic distribution of note values generated by the micro GEN1.

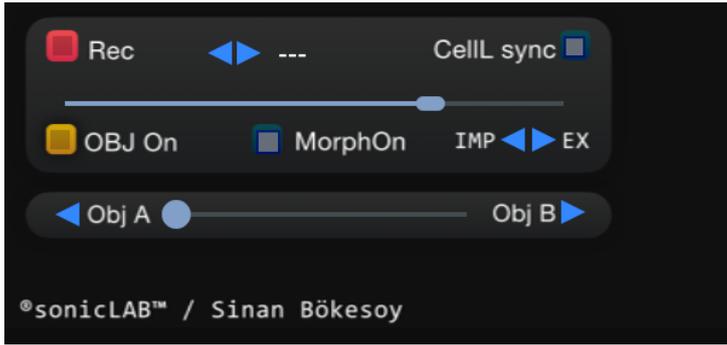
You can store 12 different patterns along with a preset. Just select the corresponding slot to edit it further.

Scale quantizer is available to be applied after the note pitch calculation for each micro event. Currently major, minor, pentatonic and whole tone diatonic scales are available. The input midi note to the plugin defines the root key as well.

We have developed a unique visualisation for these deterministic patterns in *Cosmosf* style. No explanations further, you will discover it while inputting different values to your pattern.



introducing the “Object engine” in *Cosmosf*



Sound as an ‘object’ formed by organized elements, is exhibited in continuous motion and a process of change. *Cosmosf* introduces various tools to generate the elements of the sound, sets them in motion between stochastic/deterministic and ordered/disordered frontiers.

In order to add a new level of sonic design, the “Object” in *Cosmosf* lets you drive a gigantic number of gesture recording tracks set for each of *Cosmosf* parameter. All these tracks form a sequencer which you can drive in various ways.

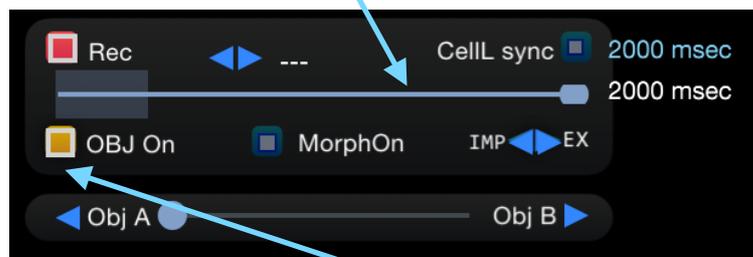
First you set a time cell of the Object duration, hence the length of the recording in msec. by tapping on this text button.



This slider will slow / speed up the Object motion. According its setting, the actual length of the object is displayed here.

For example if the slider is at half value, then the actual will be two times longer then your time cell setting.

Now you can start to record some gestures to the Object sequencer. For this you do activate the recording mode, either by clicking this **Rec** button or the ‘r’ key on your computer keyboard.



When you activate the Recording mode, it automatically turns on the Object mode as well. Then if *Cosmosf* is running, a rising bar indicates the relative time state of the sequencer which indeed is always in looping mode.

While in recording mode, you can move the sliders or click on the button switches of the *Cosmosf* interface, to record these changes into relevant tracks of the Object sequencer. When you touch any parameter on the interface, the relevant sequencer track gets activated and will be listening to your gestures on that parameter control.

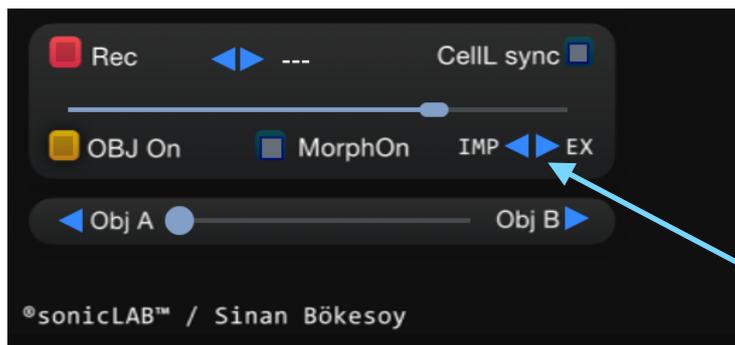
Since this is a looping recording, when you are done with your gesture performance, either click on the **Rec** button or press the 'r' key on the keyboard again to deactivate recording. Otherwise each new loop in the recording mode will override the previous recorded values.

As soon as you deactivate the recording mode, the Object sequencer track switches to playback mode and plays your latest track recording on the relevant parameter control. Even when you enter to recording mode again, the previously recorded tracks will be in playback mode unless you touch the relevant parameter on the *Cosmosf* interface.

You can record multiple gestures with your Midi controller at the same time, and each gesture will be recorded to its relevant track of the Object sequencer.

Click on the 'e' key on the keyboard to erase the current Object sequencer as a fresh restart.

Click on the Obj On/Off button to deactivate the Object sequencer playback.

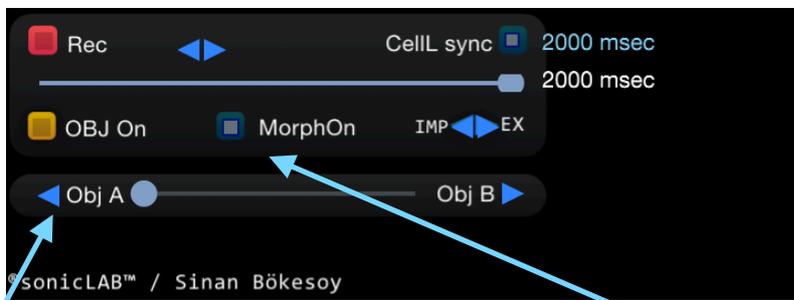


You can export your recorded Object sequence to the haddisk of your computer with EX button and import an existing one from the haddisk with the IMP button.

You can save the state of the Object sequencer along with its sequence data filename to each preset of *Cosmosf*. Once you recall that preset, the relevant settings will be done and the sequence file will be loaded to the Object sequencer.

If there are MBot instances connected to the M31 on your project scene, then during its playback the Object engine will share its performance data with the MBots as well. Likewise all MBot instances will accompany to the M31.

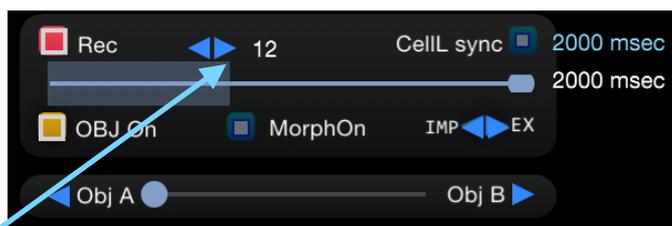
Other parameters on the Object panel are;



Morphing between two Object sequences is possible. You can load two independent sequences to locations **ObjA** and **ObjB** and then activate morphing by clicking on the **MorphOn** button.



The morph slider between ObjA and ObjB will set the degree of morphing.



You can even quantize a running Object sequence by setting the quantize value the same manner as on a DAW with this button. The time bar will also behave in stepwise motion accordingly.

You can set the Object length the same as the macro cell length of *Cosmosf* by activating the **CellL sync** switch. Thus they will synchronize easily.

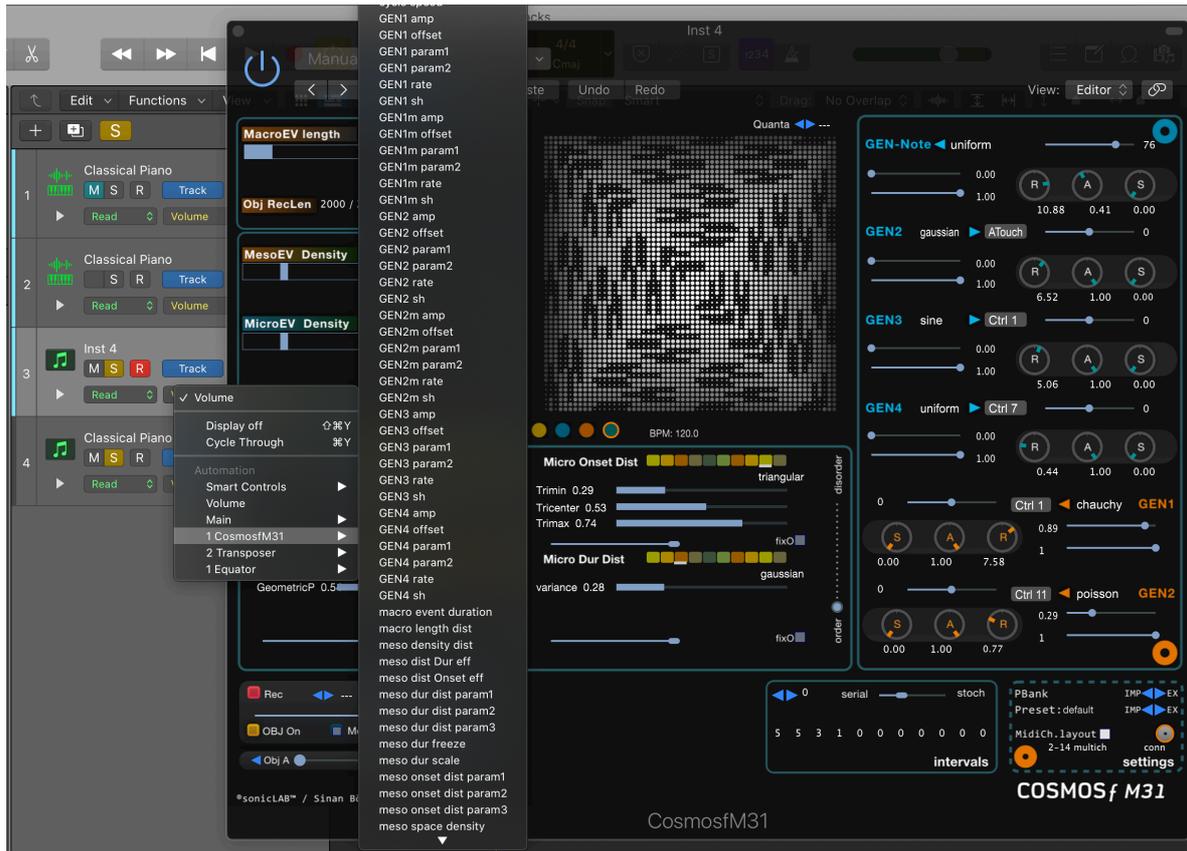
As said, time stamping of dense MIDI events such as the *Cosmosf* M31 generates needs also taking careful measures. At the moment *Cosmosf* calculates all the events at the start of each macro event, and most of the parameter changes on the UI will be applied on the next macro event cycle.

Therefore, to get out most of the Object Engine within *Cosmosf* M31, we suggest you to choose short MacroEvent Length's, so that your gestures can be frequently mapped to the following macro event cycles. Your Object Engine record buffer length can be much longer then the MacroEvent Length.

automation of Cosmosf parameters in DAW

Cosmosf M31 has many parameters which you are able to automate with the DAW, which would give you great possibilities to shape and control the outcome of Cosmosf in realtime and continuum on many levels.

Below you can see a Logic screen, and the parameter list of Cosmosf to be assigned on the relevant track. Different DAW applications have different methods for doing this.



Preset import / export



You can import and export presets of the selected Preset Bank by clicking on the appropriate button. When you select IMP, a window of 16 preset slots will show up and by clicking on it, you can load the preset you wish to run / edit.

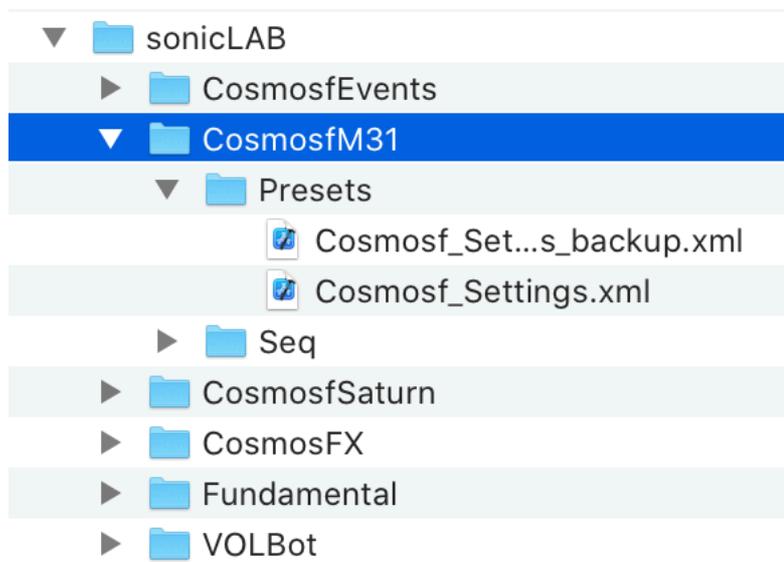
You can rename the preset by typing on the preset name field. And you can export your preset on any slot you wish by clicking on the “EX” button.

The preset export operation is decisive and alters the current bank preset slot. So in order to create a new project bank, we suggest you to create a copy of your departure bank on the *sonicLAB/CosmosfM31/Presets* directory inside your *User's Shared* folder. (OSX) On Windows, it would be inside the *Documents/sonicLAB/CosmosfM31* folder.

Cosmosf applies the preset export operations always on the latest loaded bank.

Cosmosf will create a backup copy of its default bank inside this directory automatically during the installation.

You can export the current loaded bank with another name by typing its name on the PBank name field and then exporting the bank as a file by clicking on the EX button.



Cosmosf M31 and MBot : a fleet of Bots.

Cosmosf MBot is identical to M31 but it is without a UI. And the Cosmosf M31 acts as a server and transmits its parameter data to all existing MBot's on other tracks in your DAW project.

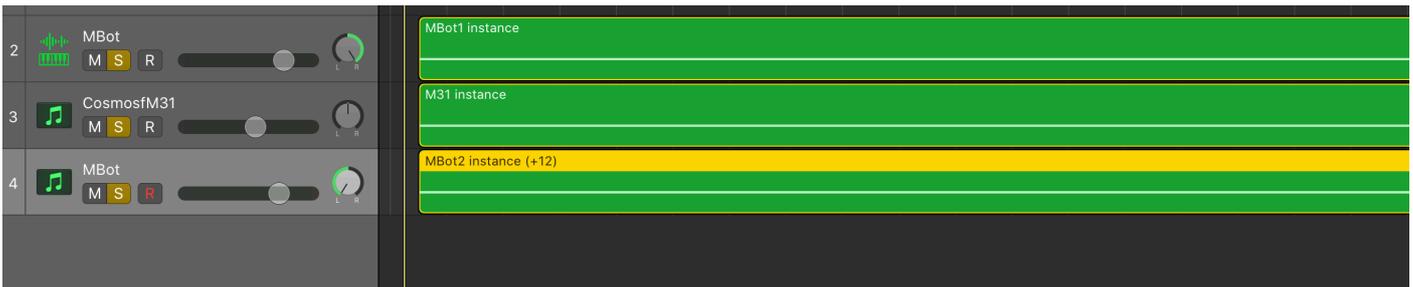
The BOT concept of sonicLAB is unique and has been initiated with the sonicLAB VOLBot and EnVBot.

The server and the bots share the same parameter space, e.g. user settings of parameters. But since they run independent stochastic iterations, all these instances offer a different version, an alternate reality.

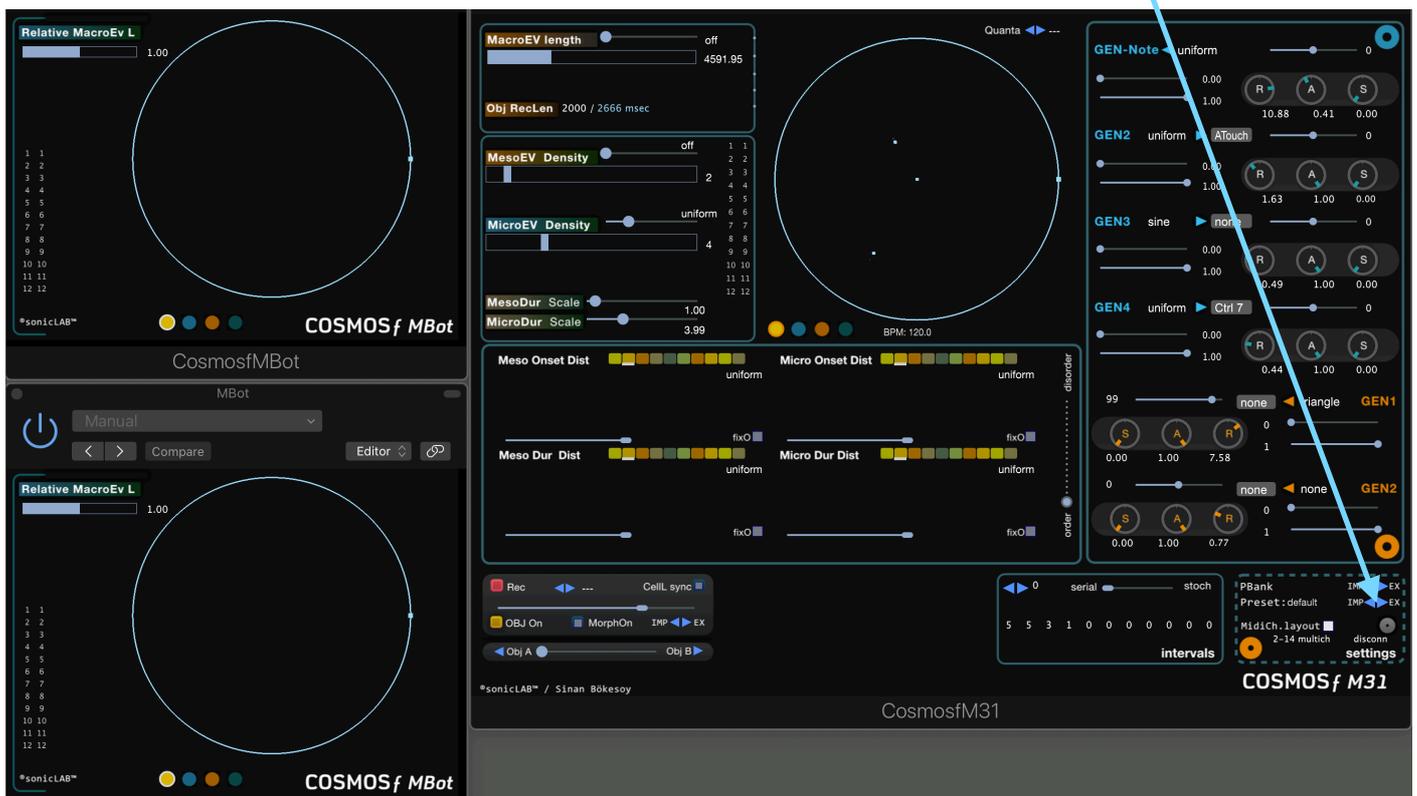
In addition, the Cosmosf MBot's have still their own parameters which can make them for example running with a different speed in comparison to the M31. Let's have a look.

Initialization of a basic Cosmosf M31 and MBot setup on your DAW:

We can directly start with an example case , we choose Logic for the DAW. Below you can see that two tracks have MBot instances and one track in the middle has the M31. Each have the same midi track (one note input only) starting together.

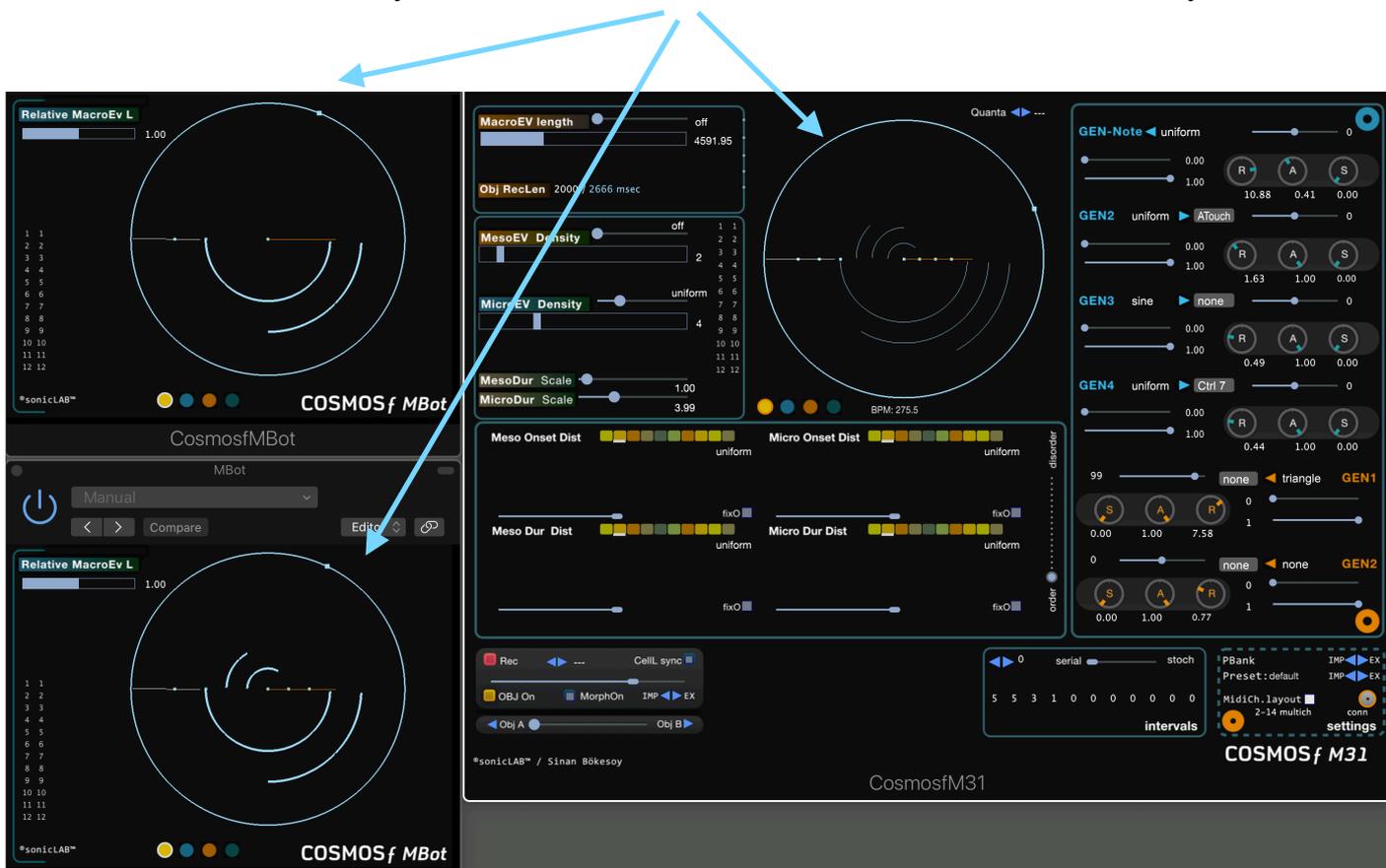


The MBot's are not connected to the M31 initially. For this we have to click on the connection button to launch the connection process.



As soon as you click on the connection button, M31 will find the existing MBot's and will establish a connection. (The label will show "conn").

We start Logic and as soon as the song pointer reaches the note event, all parameter data of the M31 will be shared with the MBots and as you see below, their event distribution has initiated successfully.



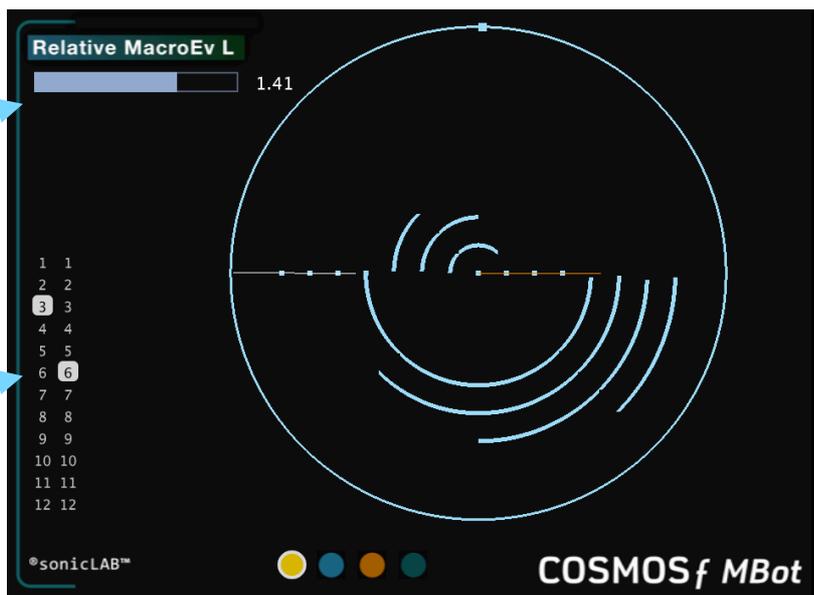
The MBot has these distinctive parameters on its own :

The Relative Macro EvL slider :

The value set here will be multiplied with the MacroEvent Length of M31 and will be assigned as the alternate Macro EvL which this MBot will use.

Event mute switches :

The M31 will not share its event mute switch states with the MBots, because they have their own. This is useful to spread different meso / micro events among each Cosmosf instances in your project.



MBot's have no preset mechanism, their current settings will be saved along with the DAW project.

While you are working on your project and need to add more MBot's, then after adding them you need to disconnect the M31 and connect again, so that it can refresh the new state of MBots.

The existing MBots in your project can be connected to only one M31. But you can use multiple M31 instances in your project.

We have prepared example projects for new users of Cosmosf M31 and MBot. These Logic and Live projects will help you understand quickly how to setup the M31 and MBot on these DAW's.

Please check them at this folder; you will find also a *readme* file describing each project.

https://www.sonic-lab.com/Cosmosf_files/M31_projects/

published articles

BOKESOY, S. ; PAPE, G. « Stochos: Software for Real-Time Synthesis of Stochastic Music» in : *Computer Music Journal* 27(3) MIT Press 2003. - pp. 33-43.

BOKESOY, S. « The Cosmos model : An event generation system for synthesising emergent sonic structures » in : *Proceedings of the International Computer Music Conference*, Barcelone, pp. 259-262, 2005.

BOKESOY, S., « Feedback Implementation within a Complex Event Generation System for Synthesising Sonic Structures » in : *Proc. of Digital Audio Effects (DAFx'06)*, Montreal, Canada, pp. 199-203., 2006.

BOKESOY, S., « Synthesis of a Macro Sound Structure within a Self-Organising System » in : *Proc. of Digital Audio Effects (DAFx'07)*, Bordeaux, France, September 10-15, 2007.

BOKESOY, S., « Presenting CosmosF as a Case Study of Audio Application Design in Openframeworks» in : *Proceedings of the International Computer Music Conference*, Ljubljana, September 9-14, 2012.

BOKESOY, S., «The development of a morphing tool in Cosmosf » in : *Journées Informatique Musicale*, Paris, May 12-15, 2013.