**Synfire User Manual** 

# Contents

apter 1. Concepts	1
Music Prototyping	1
Parameters	
Figure	
Figure Recognition	5
Figure Recording	
Harmony Concept	
Fundamentals	
Keys	9
Scale Sets	
Harmonic Context	
Progressions	14
Harmonization	15
Polytonality	
Catalog	
Phrases	17
Rendering	
Arrangements	
Tracks	20
Containers	21
Global Parameters	
Embedded Library	
Arrangement Rack	
Snippets	24
Palettes	
Libraries	
Factories	
Sketches	
Audio System	
Instruments	

Devices	
Racks	
Audio Engine	41
MIDI Routing	
Cognitone Transport	
Drones	
User Interface Basics	
Mouse	
Parameter Outlet	
Multiple Selection	
Crash Reports	
Chapter 2. User Interface	
Sidebar: Rack Module Library	
Sidebar: Phrase Library	
Parameter Inspector	
Instrument Properties	
Container Properties	60
Notation Export Settings	61
External Synchronization	
Harmony Settings	
Figure Inspector	
Take Settings	74
Interpretation Settings	
Morphing Setttings	
Time Inspector	
Help Browser	85
Phrase Editor	86
Click Board	
Arrange App	
Structure Page	
Overview Page	
Matrix Page	

Progression Page	105
Harmonizer Page	
Palette Page	
Snippets Page	
Library Page	122
Sounds Page	
Parameter Block	143
Parameter View	144
Display Options	
Transport	146
Sketch App	147
Song App	149
Library App	149
Palette App	150
Progression App	
Catalog App	153
Audio/MIDI Setup App	
Audio Settings Page	
MIDI Settings Page	
Inputs Page	
Global Rack Page	164
Global Instruments Page	171
Devices Page	173
Repository	
Panels	
Circle Of Fifth	
Keyboard Widget	
Scale Board	
String Instrument Panel	
Harmonic Context Panel	
Dialogs	
Preferences	189

	Scale Selection Preferences	
	Sound Assignment	
	Custom Controllers	
	Custom Articulations	201
	Online Updates	
	Import MIDI File Dialog	
	Keyboard Shortcuts	210
Dro	ne	211
Chapter	3. Factory Documentation	213
Eler	nents	213
	Metrics	
	Similarity	
	Spans	214
	Types	215
	Segments	217
	Step Distribution	
	Beat Divisions	
Par	ameter Factories	
	Rhythmic Bass	
	Metric Stops	
	Rhythmic Chords	
	Static	
	Functional	
	Chord Machine	
	Stray Chords	
	Metric Pauses	
	Alternating	226
	Coordinated	
	Bass One	
	Random (Flat)	
	Sequencer	
	Random (Structured)	

Syncopes	
Metric Velocities	227
Static Velocity	
Line Factories	
Bass (Multi)	
Bass (Single)	
Piano Bass	229
Arpeggio Chords	
Guitar Chords	
Piano Chords	
Arp One	
Chained Elements	
Generic Melody (Single)	
Generic Melody (Multi)	
Piano Right Hand	
Question + Answer	230
Develop A	230
Vocals	230
Phrase Factories	
Counterpoint Melodies	
Counterpoint (Skipped)	230
Piano Parts	231
ExperimentalKit1	
Chapter 4. Tasks	
Audio Setup	232
Assigning Sounds	
Setting Up MIDI Input	
Creating Device Descriptions	234
Configuring Rack Modules Manually	237
Synchronizing With a DAW	238
Relocating From Engine to DAW	
Inspecting The Audio Engine	

Editing	
Playing Palettes	
Moving Parameters	242
Editing Parameters	
Optimizing Phrases	
Harmonizing a Phrase	254
Snapshots	
Fine Tuning Interpretation	
Fitting Music Between Cue Markers	
Recording	
Recording Parameters	
Live Chord Detection	
Recording Snippets	
Making a Sketch	
Import, Export, Printing	
Limitations Of Import	
Import SMF	
Export SMF	
Export From Drone	
Export Notation	
Export Audio (Bounce To Disk)	
Drag & Drop Export	
Reusing Containers	
Printing	
File Management	272
Installation	273
Backup	
pter 5. Workflows	
Strategies	278
Songwriting	
Collecting Phrases	
Collecting Harmony	

Production vs. Notation	
Audio Engine vs. DAW	283
Building Structure	
Chapter 6. Tutorials	285
Palette Surfing 1	
Palette Surfing 2	286
Palette Surfing 3	
Palette Surfing 4	
Chapter 7. Troubleshooting	288
Graphics Glitches on Windows	
Plug-ins Can't Access Sounds	289
State Of Drones Not Restored After Opening	
Unpredictable Auto-Chords Output	
Sound Allocation Monitor	
Chapter 8. References	291
Encyclopedia of Parameters	291
Analysis	291
Chromatic	291
Controllers	292
Dynamics	293
Figure	
Flow	298
Harmony	
Interpretation	
Layer	
Length	
Lyrics	
Morphing	305
Output	305
Pause	306
Preview	307
Rhythm	

Scheme	
Shift	
Skip	
Step	
Sustain	
Take	
Tempo	
Time	
Transition	
Transpose	
Variation	
Velocity	
Volume	
Harmonic Functions	
Roman Numerals	
Parameter Conversion	
Syntax	
Synchronization Settings For DAWs	
Feature Comparison	
Glossary	
Chapter 9. Boilerplate	340
Software License	
Copyright	

# Chapter 1. Concepts

# Introduction to the terms and fundamentals of Synfire

Synfire uses powerful AI techniques and sophisticated knowledge-based algorithms to understand and process music at a high level of abstraction. Learn about the general structure of Synfire and the concepts of music theory it is based on in this chapter.

# **Music Prototyping**

# What it's all about

Composing original music from scratch can be a challenge because the longer you work on a project, the more difficult it becomes to make substantial, sweeping changes. Thus, more often than not you find yourself settling for a mediocre compromise because starting over would be too much effort. That's a shame, because this way you fall way short of your creative potential.

The idea of **Music Prototyping** is to render music dynamically from reusable components that you can edit and replace until you eventually decide your work is done. And even beyond that point, you can still make far reaching changes with ease.

- Build music from elements that can be generated, sketched, collected and combined in ever new ways.
- · Listen to results immediately and assess them more objectively.
- · Get inspiration from many happy accidents and surprises.
- Master all the instrumental parts that breathe life into your music.
- Discover musical styles beyond your routine and habits.
- Escape writer's block.

Music Prototyping can also break up your routine in an exciting new way and give rise to fresh ideas and new motivation.

# **Rendering Music From Models**

The rendering of **Music Prototyping** works somewhat like Computer Generated Imagery (CGI), where individual pictures (frames) are generated from 3D models, textures and light sources. Likewise, Synfire renders MIDI output from models by mapping <u>Figure (on page 2)</u> against <u>Harmony (on page 300)</u> and many other dynamic parameters. Altering any of these parameters leads to different music output.

# **Any Style**

Some popular music apps are custom-tailored to a specific style of music and put an emphasis on entertainment value and instant gratification. They are fun to use, but the music pretty much sounds all the same.

Synfire is not hard-wired to a specific style and workflow. It is an extremely deep and powerful tool. We could have flattened the learning curve by baking in a lot of assumptions about workflow and style, but we didn't want to do that at the expense of artistic freedom. Take your time to make yourself familiar with this powerful tool. It will get second nature sooner than you might think.

# **Your Decisions**

Synfire can inspire you with a permanent stream of new ideas. The ultimate creative decisions however are yours to make: Which style to choose, which of the many emerging elements to keep or discard, which path to take from first draft to finished score. And most importantly, to learn and understand the intricacies of your particular genre.

# Parameters

## The components music is rendered from

**Parameters** are somewhat like automation lanes in a DAW, although their various shapes and functions go far beyond that. Each parameter has a distinctive look and takes on a specific role in the process of rendering music output.

Figure (on page 2) and Harmony (on page 300) are the most visible and influential parameters, but there are many more. Some add subtle variation and expression. Others radically transform your music into something else. Find an extensive list of all parameters in the Encyclopedia of Parameters (on page 291).

### Note:

Parameters are filled with **Vector** data that you can view and edit. To keep things simple, we don't formally distinguish between a parameter and the data it is filled with and rather use the term **Parameter** (*on page* 2) for both.

#### Loops

Unless looping has been turned off with *Parameter > Repeat*, the parameters of a **Phrase** repeat independently of each other for the duration of the <u>Container (on page 21)</u> they are in. Thus their influence may evolve separately over time. For instance, a short **Figure** may be accompanied by a long and evolving **Velocity** parameter.

Containers themselves can't loop, but by resizing them, the phrases inside will loop for that duration.

#### **Related information**

Rendering (on page 18) Encyclopedia of Parameters (on page 291)

# Figure

### The parameter that provides musical expressions

#### Video Tutorial

Synfire introduces a format for writing music that is neither MIDI nor audio, nor traditional notation. It roughly resembles what improvising musicians might have in their head: Fragments and melodic movements, broken down into small connected units. It can be placed anywhere to produce different music depending on its surrounding. In fact, it's primary purpose is to be reused this way all the time.



### Video Tutorial

The Figure (on page 294) parameter carries a parametric format of musical expression (on page 2) that is independent of pitch and harmony. It contains **Symbols** grouped into **Segments**, the latter of which are responsible for preserving melodic movement and expression, regardless which harmony a Figure is rendered against.

### Important:

The most important thing about Figure is that it's not a substitute for notation or piano rolls. It's an **algorithm**.

## Note:

The term **Figure** stands for a technical data structure here. Not to be confused with the musical notion of a melodic shape.

### **Segments**

One or more **Symbols** are grouped to form a **Segment**. Segments denote musically meaningful units that are supposed to preserve their relative shape. Symbols in a segment may be stacked vertically (chord), placed in a row (melody), or any combination of the two.



Segments are selected, edited and transformed as a cohesive unit, making them a convenient device for building melodies and musical expressions of arbitrary complexity.

## Anchor

Every segment has one **Anchor** that denotes a point of emphasis that marks the melodically most important symbol of a segment. The anchor is rendered first and then other symbols are rendered relative to it. It also determines the position of the segment on the timeline.

# i Tip:

If you want a segment to end on a particularly important target note, make that last symbol the anchor. Likewise, if you care most about any other note in between, make it the anchor.

# Symbol Types



Symbols are notated on a staff with horizontal lines that, for most symbol types, correspond to the steps of a scale. The bold middle line is a reference that depends on the type of symbol.

Н

**Horizontal**: Best for long running melodies relative to the current key. The zero line denotes the root of the **Reference Scale** (on page 11) of the **Global Key**.

V

**Vertical**: Best for ornaments, licks or improvised melodies. The root of this scale changes with every chord. The zero line denotes the root of the <u>Vertical Scale (on page 8)</u>.

С

**Chord / Arpeggio**: Best for chords, horn stabs, arpeggios and finger picking patterns. The zero line denotes the lowest note of the current chord in its current inversion and voicing. Each line represents a note in the chord.

I

**Interval**: The anchor describes an interval of the chord. All other symbols are mapped relative to the anchor. Use this type to start a segment on a desired chord interval and let it continue from there.

For example, 3 maps to the third, 5 to the fifth, and 7 to the seventh (since the grid starts at zero, it is one off). Negative values go in the opposite direction. Since -3 is a third below root, it is no longer actually a third. We recommend you write positive anchors to avoid confusion.

#### Ρ

**Pitch**: Absolute pitch. For drums, percussion and other sounds that need not follow harmony. The zero line denotes C4.

В

**Bass**: The zero line denotes the bass of the current <u>Harmonic Context (on page 12)</u>. Each line is a step on the Vertical Scale.

R

**Relative**: The Anchor is mapped like a **Chord** symbol, while additional symbols are rendered like **Vertical** symbols relative to the anchor. Use this type to start a segment on a desired chord note and let it continue from there.

#### Gray

Exclusively used by the Take (on page 312) parameter. Absolute pitch.

The best way to make yourself familiar with the different symbol types is to draw a segment in a short **Figure** and let it repeat over and over while a chord progression moves on.

# Note:

Segments may contain only one type of symbol. Types cannot be mixed within the same segment.

### Note:

The **Vertical** type is available with the Express and Pro editions. The **Relative** and **Interval** types are available with the Pro edition only.

### **Related information**

Figure Recognition (on page 5)

# **Figure Recognition**

### Separating musical expression from harmony

Recorded or imported static MIDI notes need to be translated to dynamic <u>Phrases (on page 17)</u> in order to be useful for <u>Music Prototyping (on page 1)</u>. To accomplish this, musical expression is separated from harmony and the resulting phrase is annotated with hints, so a subsequent reversal of this process can be as accurate as possible. Every step in this process is ambiguous because the required information is nowhere to be found and needs to be guessed from context.

- 1. Estimate keys, chords and scales underlying a MIDI take.
- 2. Estimate the exact positions of chord changes.
- 3. Remove inaccuracies without sacrificing expression.
- 4. Identify separate voices.

- 5. Guess a useful grouping of melodic fragments, chords, bass lines.
- 6. Add hints where a phrase might encounter difficulties when it is rendered against different harmony.
- 7. Clean up and optimize the result.

For the average recorded take, there are millions, sometimes billions of possible solutions to figure recognition. Only a few hundred are musically plausible. These need to be assessed and sorted out. To accomplish this, Synfire draws on an extensive knowledge base supported by artificial intelligence algorithms.

### Limitations

Due to the ambiguity and guessing that is involved, figure recognition can't possibly be accurately reversible. That is, a Figure (on page 294) rendered against its own estimated Harmony (on page 300) will most likely produce MIDI notes that are slightly different from the original.

This is no fault of the algorithm, but a general limitation when music is approached with mathematics. Music is not an exact science, but an artifact of human culture. There is only so much a software can do to formally represent and process all aspects of music.

Figure recognition is still good enough to be useful more than 80% of the time. It is a fantastic tool for collecting reusable phrases from recorded performances. After all, <u>Music Prototyping (on page 1)</u> is about making new original music, not faithfully recreating existing compositions.

# **Figure Recording**

Figure (on page 294) can be recorded by capturing incoming MIDI messages into the Take (on page 312) parameter and then running Figure Recognition (on page 5) on it.

### **Related information**

Recording (on page 260) Recording Parameters (on page 261) Setting Up MIDI Input (on page 233)

# Harmony Concept

# About scales, chords, keys and everything that constitutes harmony

In order for a Figure (on page 294) to produce MIDI output, it requires at least a context of Harmony (on page 300) to provide a frame of reference for its relative symbols and segments. The following sections introduce the basic concepts underlying the Harmony (on page 300) parameter of Synfire.

# **Fundamentals**

About notes, chords, horizontal, vertical and modal scales

### **Notes**

The term **Pitch** refers to the frequency of a tone. Music theory is mainly organized around **Pitch Class** though, which implies a pitch and all its possible octave transpositions. That is, if we talk of Bb or D# what is really meant is a pitch class rather than a frequency. Some ideas of music theory can't be explained without distinguishing these notions.

In order to keep it simple, we use the term Note synonymously for both Pitch and Pitch Class depending on context.

- 1. Note means pitch class, when we talk about chords, scales and harmony in general.
- 2. Note means pitch (frequency) with a duration, when we talk about pitch ranges, instruments and sounds.

# Note:

For display and parsing, specific naming conventions (on page 325) apply.

# Chords

When multiple notes happen to sound at the same time, we have a **Chord**. When these notes are sorted by pitch class, starting from the chord's **Root** pitch, we get the **Interval Structure** that lends the chord its name. The sound of a chord can be varied by **Inversion** (ordering of intervals) and **Voicing** (overall octave range).

Chords consist of a basic **Triad** of three notes plus optional **Extensions**. These are intervals added to make it more complex, rich and ambiguous. For example the basic triad  $\underline{Am}$  can be extended as  $\underline{Am6}$ ,  $\underline{Am7}$ ,  $\underline{Am9}$ ,  $\underline{Am11}$  or  $\underline{Am(6,9,\#7)}$  and many more.

# Note:

For display and parsing, specific naming conventions (on page 327) apply.

# **Scales**

A **Scale** is a series of **Notes** starting from a **Root** note upwards. Most scales in Western music repeat every octave. The timbre and character of a scale is determined by its **Interval Structure**, the distances between its notes measured in halftones.

Examples for scales widely used in Western music are Major, Natural Minor, Melodic Minor, Harmonic Minor. There are scales with seven steps (septa-tonic scales), or five (pentatonic), or any other number. A scale with twelve tones in halftone intervals is called the **Chromatic Scale**. It is equivalent to all keys on a piano keyboard.

Synfire distinguishes between **Vertical Scales** and **Horizontal Scales**, which are physically the same thing, but take on separate roles and are thus traditionally labeled differently. That is, the same physical scale often has a different name depending on its role.

```
E.natural-minor (horizontal) = E.aeolian (vertical)
```

# Note:

For display and parsing, specific naming conventions (on page 328) apply.

# **Vertical Scale**

The **Vertical Scale** provides notes for melodic ornaments and improvisation. For each chord, a vertical scale can be picked according to preference and style. It is named vertical because its notes are stacked vertically above the current chord. Both, the chord and the vertical scale, start at the same root. It is important to remember that the vertical scale changes with every chord.



The vertical scale determines which notes are allowed for building melodies over a chord. Some may be outside the **Horizontal Scale**, if only the duration of the chord. Vertical scales are particularly important for improvisation. A musician may play up and down the scale and be assured that it fits the chord, although that alone doesn't make a good melody, of course.

Countless vertical scales exist, originating from various eras, cultures, and music genres. A selection of the most commonly used scales is included with the <u>Catalog (on page 16)</u> of Synfire, to which you can add more scales as needed.

# **Horizontal Scale**

While the vertical scale is tied to a chord and therefore constantly in motion, the **Horizontal Scale** is meant to continue unchanged for a longer period of time. It is strongly tied to a **Key**, denoting its steps I, II, III, IV, V, VI, VII (German: "Tonleiter"). Melodies built on the horizontal scale move across multiple chords more evenly, even though the vertical scales over each chord may temporarily extend or reduce the set of allowed notes.



A horizontal scale starts with the root note of the **Key**. It belongs to the <u>Scale Set (on page 11)</u> of the key. Traditionally, only a few scales are used as horizontal scales. These are the well-known major and minor scales and their variants.

# Mode (Modal Scale)

Some scales are used both as vertical scales for improvisation and melody construction (with jazz, in particular) as well as horizontal scales. These so-called "church modes" were introduced with modal music early in music history and are still widely used.





Modes are basically rotations of major or minor scales and are therefore physically equivalent with them although *not* identical.

Major:		Minor:	
Major	Natural minor	Melodic minor	Harmonic minor
1. Ionian	1. Aeolian	1. Melodic	1. Harmonic
2. Dorian	2. Locrian	2. Dorian b2	2. Locrian 6
3. Phrygian	3. Ionian	3. Lydian augm.	3. Ionian augmented
4. Lydian	4. Dorian	4. Lydian b7	4. Dorian #4
5. Mixolydian	5. Phrygian	5. Aeolian major	5. Phrygian dominant
6. Aeolian	6. Lydian	6. Locrian #2	6. Lydian #2
7. Mixolydian	7. Locrian	7. Super locrian	7. Altered dominant bb7

The modes listed here are only a small selection of the built-in catalog. In addition, there are the corresponding modes of melodic major, harmonic major, hungarian major, and hungarian minor.

Modes that arise from arbitrary horizontal scales are computed by Synfire only as needed and are automatically assigned a suitable name.

Related reference	
Syntax (on page 325)	
Related information	
Scale Sets (on page 11)	

# Keys

# About traditional keys, tonal center and key changes

**Key** is a fundamental concept of tonal music, which developed during a certain era and still predominates in Western music to this day.

Figure 2. Chords strictly built from the steps of the horizontal C Major scale



Many musicians think of key as being a scale, but keys do not have such a strict boundary. Key is a rather broad term for the harmony a piece of music is moving in. Modern music uses a lot of secondary dominants and other loosely related <u>chords (on page 7)</u> that are built from notes well outside the scale that lends a key its name. Still, a listener recognizes the key by characteristic chord changes (cadences) and melodies.

# **Key Changes (Modulation)**

A change of key is also known as **Modulation**. Synfire is comparatively skilled in the recognition of keys. However, due to the ambiguity mentioned above, it is a matter of judgment where exactly a key change is taking place. Concerning key changes, the decision is always with you, the composer.

There are widely differing notions of modulation and whether it requires preparation and affirmation. You should not get distracted by such debates. In modern music, you may jump from one key to another in an instant, provided it makes sense to an unsuspecting listener. And that is a matter of experimentation and testing.

With some jazz styles key changes are so frequent, a "key" printed on a score is merely a means to keep it all meaningful and readable.

# Note:

Every <u>Harmonic Context (on page 12)</u> in a progression has a key assigned. If the key of a context is different from the previous, you have a key change. It's as simple as that.

# i Tip:

As a general rule of thumb, if you change key, don't do so in the middle of something, but rather between clearly cut sections. Also you shouldn't change too many things at once, so there is something a listener can hold onto while your continents are drifting. You get the idea.

# i Tip:

After your progression moved from one key to another, you should emphasize a few chords that are unique in the target key.

# **Tonal Center**

The **Tonal Center** is what *you* assume is the root of the Major/Minor key you currently work in, regardless of the Scale Set you might use to browse interesting chords. It determines the coloring and spelling of chords and scales and somewhat influences the scale selection preference for progressions.

The tonal center lends a functional meaning to all chords. The term was coined in classical music theory, but can be a useful tool for exploring key changes in any musical genre.

#### **Related information**

Circle Of Fifth (on page 183)

# Scale Sets

#### Multiple scales bundled together

The notion of <u>Scale Set (on page 11)</u> was introduced by Cognitone specifically for **Music Prototyping** in order to support a computational model for the theory of harmony.

A <u>Scale Set (on page 11)</u> bundles one or more scales, all of them starting from a common root note. Synfire resembles traditional Major/Minor keys with predefined Scale Sets, the basis for the so-called **Standard Palettes** that you can navigate in the <u>Circle Of Fifths (on page 183)</u>.

#### **Major Scale Set**

major + harmonic major + melodic major

### **Minor Scale Set**

natural minor + melodic minor + harmonic minor

#### **Reference Scale**

The first scale in a Scale Set is called **Reference Scale**. It is the one most frequently used for building chords and melodies. It influences the order of additional scales and the spelling of notes. For example, whether a note is written Eb or D# depends on the reference scale.

### **Additional Scales**

The additional scales are thought of being variations of the reference scale, enriching the harmonic material with additional notes. You can make every additional scale become the reference scale, using the **Palette Editor** (*on page* <u>108</u>).

#### Note:

Synfire automatically creates additional Scale Sets as needed, using its knowledge of relationships between scales, chords, and keys.

# Note:

Every Palette is based on a Scale Set.

### **Related information**

Scale Set Syntax (on page 329) Editing Palettes (on page 108) Fundamentals (on page 6)

# Harmonic Context

# Information on key, chord, scales and bass that governs every moment in time

Every box in a <u>Harmony (on page 300)</u> parameter represents a **Harmonic Context**. It provides the necessary harmony information and rules all instruments must follow to render MIDI output. In most cases you just provide a **Chord** and Synfire figures out the rest. You can then alter its properties at any time.



### Key

Key (on page 9) signature as it appears on printed notation.

### Relation

The key we would have to assume if the vertical scale and chord were required to be included in the <u>reference scale (on page 11)</u> (which they not necessarily are). It provides a hint how much the vertical scale deviates from the reference scale.

### **Horizontal Scale**

The Horizontal Scale (on page 8) that defines the steps I, II, III, IV, V, VI, VII of the key.

### **Vertical Scale**

The Vertical Scale (on page 8) for building melodies and ornaments over the chord.

### Chord

The <u>Chord (on page 7)</u> selected for this context. Options like a preferred **inversion** and **voicing** can be set, although <u>Interpretation (on page 301)</u> settings per instrument decide whether to follow these default preferences.

#### Bass

Desired bass interval, which does not necessarily have to be contained in the chord. The bass is also independent of the chord inversion, i.e. it is not necessarily the lowest note of the chord.

#### **Harmonic Functions**

Possible role(s) the chord may take in a progression according to the <u>Theory of Harmonic Functions (on</u> page 319).

### Numeral

Roman numeral of the chord within the key.

#### **Extensions**

List of <u>chord extensions (on page 7)</u> possible under the current scales.

#### Annotations

The way a chord name is displayed tells you something about it.

#### **Italic Font**

The harmonic context is for one of the multiple layers of a polytonal progression.

#### **Exclamation Mark (!)**

Indicates one or more of the following mismatches:

- The horizontal or vertical scale violate current <u>Scale Selection Preferences (on page 192)</u>. For example, when the vertical scale is not a subset of the horizontal scale.
- 2. Chord notes or bass are not included with the vertical scale.

This is not a problem, but may lead to dissonances that are not approved by the scale selection preferences. You can remedy this by selecting different scales, changing the scale selection preferences, or by re-estimating the whole progression, or parts of it.

#### **Question Mark (?)**

The harmonic context is incomplete or broken. This should happen rarely if at all. You can remedy this by setting different scales or by re-estimating the progression, or parts of it.

### **Related information**

Harmony Settings (on page 69) Editing Harmony (Progressions) (on page 250) Scale Selection Preferences (on page 192)

# Progressions

# The succession of harmonic contexts over time

## Video Tutorial

A <u>Progression (on page 14)</u> consists of one or more <u>Harmonic Contexts (on page 12)</u> lined up sequentially on a timeline. It is what gets assigned to the <u>Harmony (on page 300)</u> parameter. Among musicians it is often called "chord changes", "chord progression", or simply "chords".

Ebm s	Gbmaj7(9) tG, sP (t tP dG SSS)	Ab(add9) dP, SS	<b>Bmaj7</b> (9) N, sG (s tG sP)	<b>Bb9</b> (T d)
1 (Eb)	1 (Gb)	1 (Ab)	1 (B)	7 (Ab)
Eb.dorian	Gb.lydian	Ab.mixolydian	B.lydian	Bb.mixolydian
Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor
Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor
Ebsus4	Eb S	Bbm t	Gbm(maj7) tg,sp	Ab7 (dP SS)
4 (Ab)	1 (Eb)	1 (Bb)	1 (Gb)	7 (Gb)
Eb.dorian	Eb.lydian-b7	Bb.melodic	Gb.lydian+2	Ab.mixolydian
Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor
Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor	Bb.Minor

A progression sets the rules for all instruments. Still, each instrument produces quite different MIDI output depending on its **Playing Ranges**, the <u>Figure (on page 2)</u> parameter, the <u>Interpretation (on page 301)</u> parameter and many more parameters. Thus, although you can configure details like bass, chord inversion and voicing for the Harmony parameter, this information can't be turned into MIDI output without a particular **Instrument** that interprets it.

### Note:

A progression cannot have gaps or be empty. It must provide a Harmonic Context for every point on the timeline, regardless whether an instrument is actually playing there or not.

# Note:

While some DAWs feature a "chord track", these often coerce MIDI notes into a rigid grid, which only works for simple MIDI sequences and a limited range of musical styles. Unlike Synfire, a DAW is not able to recognize and re-compose phrases in a musically meaningful way in order to follow an arbitrary chord progression. Dealing with harmony in an intelligent way is a major strength of Synfire.

#### Use

Depending on who you ask, the significance of chord progressions is often overestimated. After working with Synfire for a while, you will notice how replacing Harmony affects the mood and listener experience of a composition or song. Sometimes it's subtle. Sometimes it's very noticeable, but it rarely makes or breaks a song. Instrumentation, rhythm and arrangement also contribute their part. It's all of them together that make the music and neither can be judged in isolation.

This is why **Music Prototyping** is so priceless. It helps you judge your work from a comfortable distance, all components open to change at any time.

https://en.wikipedia.org/wiki/Chord\_progression

#### **Related information**

Harmony (on page 300) Progression App (on page 152) Progression Page (on page 105) Editing Harmony (Progressions) (on page 250) Controlling Inversion, Voicing and Octave (on page 254)

# Harmonization

# Estimating keys, scales and chords from MIDI input

The process of estimating keys, scales and chords from static MIDI input is called **Harmonization**. It uses a <u>Take (on</u> <u>page 312)</u> parameter for input and creates a <u>Harmony (on page 300)</u> parameter as output.

Harmonization is an important part of **Figure Recognition** (*on page 5*), which separates musical expressions from harmony in order to create a reusable **Phrase** that can be rendered with different harmony.

The <u>Harmonizer (on page 105)</u> now is a tool for you to control the conversion from Take to Harmony manually and step by step.

#### **Related information**

Harmonizing a Phrase (on page 254) Harmonizer Page (on page 105)

# Polytonality

## Superimposition of multiple chords

Synfire supports the superimposition (layering) of multiple chords over each other. Although <u>Harmony (on page 300)</u> is a **Global Parameter**, it may consist of up to four layers, each of which can be assigned to a different instrument using the <u>Layer (on page 302)</u> parameter.

Layers are created and configured with the **Progression Editor** (on page 252).



This feature is available with the **Pro** edition.

# **Related information**

Harmony (on page 300) Layer (on page 302)

# Catalog

# Database of interval structures: The foundation of Synfire's knowledge

The **Catalog of Keys**, **Scales And Chords** (short: <u>Catalog (on page 16)</u>) defines the names, interval structures and other meta-data of all chords and scales known to Synfire. It is at the heart of the AI knowledge base and thus has significant influence not only on the choice of chords and scales, but also on dynamic scale selection and palette layouts.

You can customize the catalog to your style and preference. You are even free to develop your own system of harmony, if you like. For example, based on the twelve-tone system.

Open the Catalog App anywhere from the menu (Edit > Catalog of Chords And Scales).

Types of interval structure defined in the Catalog:

- 1. Chords (on page 7)
- 2. Vertical Scales (on page 8)
- 3. Horizontal Scales (on page 8)
- 4. Scale Sets (on page 11)

Some scales may take on both a vertical and a horizontal role. Traditionally, different names are used for practically equal scales according to their role, for example the **natural-minor** horizontal scale is referred to as **aeolian** when used as a vertical scale.

# Note:

This feature is available with the Express and Pro editions.

# **File Attachments**

When an arrangement, palette or library is saved to disk, all the Catalog items it uses are saved as an attachment together with the file. When a file is opened, any items not currently in the user's Catalog will be added temporarily so the file can work. Consequently, otherwise familiar chords may show up with different names in palettes and progressions, if your current Catalog happens to include an exotic harmony system.

# Note:

Your current Catalog always supersedes any items loaded from file attachments. That is, only if an interval structure is not currently known in your Catalog, it will be added temporarily to make the file work.

#### **Related reference**

Syntax (on page 325)

### **Related information**

Catalog App (on page 153)

# Phrases

# A bunch of parameters bundled together

**Phrase** is a structure that bundles one or more <u>Parameters (on page 2)</u> so they can be copied, moved and edited as a coherent unit. Being the primary carrier of musical expression, it is the foundation of <u>Music Prototyping (on page 1)</u>. Its main purpose is to be reused in different contexts to produce different output. A well made phrase can be placed in any <u>Arrangement (on page 19)</u> under any <u>Harmony (on page 300)</u> and "just work".



Although the Figure (on page 2) parameter looks somewhat like a piano roll or fancy MIDI region, its colored symbols are relative (parametric) and thus require least a Harmony (on page 300) parameter in order to be mapped to pitch.

However, since almost *every* parameter influences MIDI output, there is no way to tell in advance which MIDI notes a phrase will ultimately play before all of an arrangement's <u>Containers (on page 21)</u> and parameters have been considered. That's why Synfire renders all of an arrangement from scratch every time you have changed something and press play.

### Important:

Phrases may carry only a single parameter, like <u>Velocity (on page 317)</u> or <u>Pause (on page 306)</u>. Keep in mind that if you move a phrase around, it will replace an existing phrase including *all* its parameters. So, if you want to copy or move a single parameter to another phrase, be sure to drag it off of a <u>Parameter Outlet (on page 45)</u> instead of grabbing the entire phrase.

Note:

The term **Phrase** stands for a technical data structure here, not to be confused with the meaning it has in classical music theory.

# Rendering

# Mapping the entirety of all parameters to MIDI output

Since almost everything in <u>Music Prototyping (on page 1)</u> is parametric and dynamic, Synfire can produce MIDI output only after all parameters in all containers have been considered. This process is called **Rendering**, during which multiple conflicts can arise that need to be solved.

In order to stay within the **Playing Ranges** of the **Instrument** while retaining melodic movements and following a voice leading strategy, it is often required to break down notes and melodies into smaller units that still sound similar to the original. You can influence this process by configuring the <u>Interpretation (on page 301)</u> parameter.

# **Parameter Routing**

Parameters influence each other in a particular way, which is illustrated in the diagram below. There are **symbolic** parameters that control music composition at a higher level and **physical** parameters that mostly manipulate MIDI output at a later stage of the rendering process.



The logic behind parameters is more complex than the diagram might suggest, but the overall sequence of computation lined out here may help you understand what happens when you edit a particular parameter's data.

# Limitations

Every time you changed something and press Play, the entire track of the instrument is rendered anew from scratch. This can be time consuming and therefore doesn't lend itself well to the real-time interactivity you may be used to from a DAW.

# Important:

Altering an already running playback does not immediately influence output. It needs to be restarted to reflect any changes you made.

# Arrangements

# About tracks, containers and everything that goes with an arrangement

An **Arrangement** includes everything needed to edit a song, composition, score or soundtrack, or whatever you are composing with Synfire.

- One or more Instruments (on page 33).
- An Arrangement Rack (on page 24) providing Sounds (on page 37) for these instruments.

- An Embedded Library (on page 23) serving as a creative clipboard for phrases and parameters.
- Lots of <u>Phrases (on page 17)</u> for the instruments, arranged in a hierarchy of <u>Containers (on page 21)</u> along the timeline.

### **Projects**

What constitutes a **Project** may vary. With most popular music styles, you probably have a single arrangement that includes all of a song, dance or ambient track. If you are composing for film, you may have multiple arrangements, each underscoring a particular scene and all of them based on one **Global Rack** with orchestral sounds. A project may also include **Libraries** with phrases you collected and optimized for a specific task.

You should keep all files associated with a project in one folder. Synfire doesn't enforce this, so it is up to you to keep your things together.

#### **Related information**

Arrange App (on page 91)

# Tracks

# All phrases for an instrument, spanning the entire arrangement

Tracks, like in a DAW, actually don't exist. There is no such thing like a track that spans the entire song. Instead, <u>Containers (on page 21)</u> are used to place <u>Phrases (on page 17)</u> for one or more <u>Instruments (on page 33)</u> anywhere on the timeline. Since containers can be nested to build an evolving structure, it is not immediately obvious what's ultimately rendered for each instrument.

Nevertheless, the entirety of phrases for an instrument over the course of an arrangement can be considered a virtual **Track**. On the <u>Overview (on page 100)</u> page, you can see the final output that your container structure ultimately renders for each track. This map aids you with navigating your arrangement in a way you are familiar with from your DAW.

### **DANGER**:

Keep phrases short! *Never* use a single phrase as if it were a long track spanning the song. Don't scroll back and forth in a huge <u>Figure (on page 294)</u>. Put multiple short phrases in movable **Containers** instead.

### Important:

Rule of thumb: When you want something to start in the middle of a container, insert a child container at that position. Don't insert blank space at the beginning of a **Figure** only to make it start at a later time, unless you really want all loops to include this long pause at every turn.

#### **Related information**

Track Sheet (on page 98) Drag & Drop Export (on page 270)

# Containers

# Place phrases anywhere on the timeline

## Video Tutorial

Containers include Phrases for one or more Instruments, including Global Parameters like <u>Harmony (on page 300)</u>, <u>Tempo (on page 313)</u> or <u>Scheme (on page 308)</u> that affect all instruments alike. Containers can be placed anywhere on the timeline and nested to build a tree structure.

- · Containers help you keep phrases short and easier to edit.
- Since every container marks some place of importance, it's a convenient locator for playback and navigation.
- Enable a non-destructive lets-see-what-happens workflow. By placing a container inside another, you can temporarily introduce a change that lasts only for the duration of the container. Move the container to apply the change elsewhere. Remove or inactivate it to undo the change.
- Help you develop a narrative of your score. Move containers with phrases around, while the overall <u>Harmony</u> (on page 300) remains unchanged. Move containers with <u>Harmony (on page 300)</u> around, while phrases stay put.

<u> 1</u>	huhuhuhu	9	huluduu	17.	ահահահահ	25	<u>uhuhuhuhu</u>	33	41		<u>mhaladadada</u>	
						Elect	ro Sprikes	Snippe	ts °			>
▽ Intro	ľ	Melody	В	Details, u	nmute	▽	С	D	Bassmania	Horns	Melody*	
Start		Part B °		Horns	Melody	⊽ CA	unmute	Melo	ody	B Details,	, unmute	
					Part B	CAA		Pa	rt B gap!		Part B	

# Structure

The Root Container at the top lasts for the duration of the arrangement.

Child containers may represent sections like Intro, Verse, Chorus, or parts A, B, C or any other sectioning that suits your style.

# 7) Tip:

A container always introduces a *temporary* change. Parameters not overridden in a container simply continue to play. When placing a container, always ask yourself: What is it that I want to change for the duration of this container?

# Important:

Rule of thumb: When you want something to start in the middle of a container, insert a child container at that position. Don't insert blank space at the beginning of a **Figure** only to make it start at a later time, unless you really want all loops to include this long pause at every turn.

## Inheritance

Parameters in a child container temporarily override the same parameter of a parent container. Only parameters for the same instrument override each other. Other parameters continue unaffected. In short: Inheritance works at the parameter level.

For Figure (on page 294) and parameters that transform it, the effect is obvious. For other parameters it can be more subtle. For example, you may override <u>Velocity (on page 317)</u> or <u>Shift (on page 309)</u> to alter the dynamics or timing of a phrase.

The inheritance of parameters is made visible through the **Parameter Trace** (on page 97).

# **Priority**

During the time span of a container, the parameters contained in it temporarily override any parameters that are already in effect. Where containers overlap, the container that starts later overrides the one that started earlier. Where containers start exactly at the same time, the one below overrides the one above.

With the Up and Down arrow keys, you can change the **priority** of a container that starts at the same time with other containers. This will change their vertical order.

## Alias

An **Alias** is a read-only shadow of a container that can be placed anywhere in order to repeat the phrases of its original without copying them. All changes applied to an original will also update its aliases. Some properties of an alias may be set independently from its original.

- Length: An alias may have a different length, playing its contents for a longer or shorter period of time.
- Inactivation: An alias may have its own active versus inactive status.
- **Inactivation of Phrases**: Individual phrases in an alias may be turned active or inactive as needed. **Pause** however is a parameter and those can only be altered in the original.
- **Priority**: An alias may have a different priority. That is, whether it overrides running parameters or not depends on its vertical place in the structure.

# i Tip:

If you need to replace phrases in an alias, put them in another container placed below it (*Container > New Parallel Container*). Use the arrow keys to change its priority until it appears below the alias. The phrases in it override those of the alias. You may want to group both containers to keep them bound together.

# Loops

Unless looping has been turned off with *Parameter > Repeat*, the parameters of a **Phrase** repeat independently of each other for the duration of the <u>Container (on page 21)</u> they are in. Thus their influence may evolve separately over time. For instance, a short **Figure** may be accompanied by a long and evolving **Velocity** parameter.

Containers themselves can't loop, but by resizing them, the phrases inside will loop for that duration.

#### **Related information**

Parameter Trace (on page 97)

# **Global Parameters**

### The special track that affects all instruments alike

The topmost track on the Track Sheet bears the name of the <u>Container (on page 21)</u>. It looks like a title placed below the <u>Harmony Lane</u> (on page 97). Any parameters (on page 2) you assign here affect all instruments for the duration of the container.

Global Parameters	D	Arrangement	
Yamaha Grand Piano	$\bigtriangledown$		

Global parameter types like <u>Harmony (on page 300)</u>, <u>Scheme (on page 308)</u> or <u>Tempo (on page 313)</u> always go here, no matter whether you drop them on a <u>Container (on page 21)</u>, a <u>Parameter View (on page 144)</u> or a <u>Track</u> (instrument name).

Still, you may also put other non-global parameter types here, like <u>Step (on page 310)</u>, Length, <u>Velocity (on page 317)</u>, <u>Volume (on page 318)</u> and many more, sometimes to dramatic effect.

### **Related information**

Track Sheet (on page 98) Parameter View (on page 144) Parameters (on page 2)

# **Embedded Library**

# The clipboard for phrases and parameters

Every arrangement owns an **Embedded Library** that serves as a clipboard of sorts for extracting, generating, collecting and reusing **Phrases** and **Parameters**. It is saved as a part of the arrangement. You can move phrases to and from other libraries (drag & drop), or save the embedded library as a standalone library file if you want to keep it for other purposes.

It basically works like a regular Library (on page 28).

Visit the <u>Library (on page 122)</u> page to browse the embedded library. There you are also able to generate new phrases using **Factories**.

Using the *Library* menu, you can open up to three other libraries at a time to show up on the <u>Phrase Library Sidebar (on</u> page 49).

### **Related information**

Libraries (on page 28) Library Page (on page 122) Library App (on page 149) Sidebar: Phrase Library (on page 49) Adding New Phrases (on page 50)

# Arrangement Rack

# Providing sounds for the arrangement

On the <u>Sounds (on page 128)</u> page you find the **Arrangement Rack** (also known as Private Rack). It is owned by and opened together with the arrangement. Its <u>Rack Modules (on page 38)</u> provide sounds for the arrangement's instruments.

Otherwise it behaves like any Rack (on page 37).



If you started an arrangement based on sounds hosted by the <u>Global Rack (on page 39)</u>, once your project is finished and ready to be archived, you should copy the global rack modules, so they are permanently archived with the arrangement. Do so from the pop-up menu on the <u>Parameter Inspector (on page 54)</u>.

#### **Related information**

Racks (on page 37) Rack Module (on page 131) Sounds Page (on page 128)

# **Snippets**

# Small containers for live performance and improvisation

**Snippets** are containers placed on a grid that is designed for live performance and improvisation. The grid's rows each present a <u>Snippet Group (on page 118)</u>.

Snippets can be started and stopped individually, although only one snippet per group can play at a time. This makes each group a set of alternative musical expressions (any parameters actually), ready for you to be scheduled live and in real-time.

Harmony	) Em9 Em9 Em(#5)	C(add2) G(add2)		
Electronic	D Electronic	D Electronic	🕞 Gap	
⊔∰ Sub Dub Bass	Sub Dub Bass	D Latin		
Popcorn 2	Popcorn 2			
ाम्म Streamer (Hold Chords	▷ Streamer (Hold C			
Spring Bass 2	Spring Bass 2			
ভ C-Bass	C-Bass			

A snippet grid can be created automatically from an arrangement, by collecting existing phrases from containers. A life performance can be retroactively converted into an arrangement.

# i Tip:

Although a snippet can contain phrases for any number of instruments, you may want to arrange your first grid such that every group is for one particular **Instrument**, while the first group is reserved for <u>Harmony (on page</u> <u>300)</u> parameters.

### Important:

**Tempo** and **Scheme** are global for all snippets and the root container of your arrangement. When you change either parameter, it affects all snippets and the root container of your arrangement alike. It is currently not possible to have snippets with different time signatures or tempos.

# Note:

This feature is available with the **Express** and **Pro** editions.

## **Related information**

Snippets Page (on page 115) Snippets Console (on page 122) Snippet Group (on page 118) Snippet (on page 120) Recording Snippets (on page 263)

# Palettes

# Surfing the space of harmony

# Video Tutorial

The Palette is a convenient tool for exploring all keys, scales and chords.



Palette layout is divided into five areas:

- 1. Middle bar with one or more Horizontal Scales. The top scale is the Reference Scale.
- 2. Chords built from the Reference Scale exclusively.
- 3. Chords built from any of remaining scales.
- 4. **Related Chords** hanging from the top. These can't be built from scales of the Palette, but still have a significant functional relation to the **Tonal Center**.
- 5. Additional Chords at the bottom edge with only a weak or no relationship to the **Tonal Center**. Such chords offer interesting transitions to other keys or can add variation to a progression.

The Palette basically is a visualization of a <u>Scale Set (on page 11)</u>. Its content and appearance can be configured with the Layout menu (*Layout > Settings* and *Coloring > Settings*).

Scale degrees can be labeled with

- Roman Numerals I, II, III, IV, V, VI, VII, VIII for Standard Palettes that denote traditional Major/Minor keys.
- Steps 1, 2, 3, 4, 5, 6, 7, 8, etc. for Alternative Palettes not limited to any type of tonality.
- Note names.

### **Chord Columns**

Chords built entirely from the **Reference Scale** appear above the middle bar. Other chords appear below. All chords defined in the <u>Catalog (on page 16)</u> are considered for display, unless you exclude them with <u>Layout > Settings</u>. Filtering chords can greatly simplify a Palette, making it easier to navigate for beginners.

## **Standard Palettes**

Standard Palettes resemble the traditional Major and Minor keys in the Circle of Fifths. They are pre-built based on Scale Sets that include the traditional horizontal scales.

Simply click on a key in the in the Circle of Fifth to select any desired key.

#### **Alternative Palettes**

Alternative palettes can be built on arbitrary scales. They are not tied to traditional Major/Minor tonality. You are pretty much free to include any type of scale in such a palette and experiment with the chords that show up. Scales may be whole-tone or half-tone scales, modes, or oriental or jazz scales with 8, 9, or more tones. You can in effect add to the **Catalog** (*on page 16*) any arbitrary interval structure as a new scale and then start a new palette based on that.

# Note:

By editing a palette, you actually edit the <u>Scale Set (on page 11)</u> it is based on. In many ways, the palette is the visual presentation of a scale set.

#### **Reading an Alternative Palette**

There is no longer a clear role for the reference scale, which can be totally unrelated to the other scales. Likewise, the root of a palette is not necessarily the tonal center. An alternative palette offers a particular perspective on a selection of chords that can be built from its scales.

This is pretty much uncharted territory, inviting you to experiment and explore. The alternative palette is a tool for navigating an otherwise huge harmonic space, following seemingly odd but meaningful relationships. What looks like a straight path through the palette may actually be a trip through several traditional keys. The logic behind it may be obscured, but it will shine through your composition and make sense. Discover your own system.

#### **Scale Selection Preferences**

Synfire prefers to select scales that can be derived from the scale set of the palette. The scale set thereby significantly influences melodies playing along with a chord. A harmonic context for the same chord will sound differently depending on the palette it was taken from.

You can influence this to some extent by editing the scale selection preferences (Edit > Scale Selection Preferences).
## **Related information**

Playing Palettes (on page 240) Editing Palettes (on page 108)

# Libraries

## Collections of phrases and parameters, the building blocks for arrangements

## Video Tutorial

A Library is a collection of Phrase Pools, each of which contains one or more <u>Phrases (on page 17)</u> populated with <u>Parameters (on page 2)</u>.

All sorts of phrases and parameters, be they imported, sketched, recorded or generated, can be collected in a library and organized in folders. The point is to have these ready for reuse by conveniently dropping them on any destination in an arrangement.

## **Phrase Pools**

A **Phrase Pool** (short: Pool) contains one or more <u>**Phrases** (on page 17)</u> that are deemed variations of a musical expression. The properties of a pool are shared by all its phrases.

- Sound
- Interpretation (on page 301)
- Scheme (on page 308)
- Tempo (on page 313)
- Preview (on page 307) (optional)

Since these properties are shared by all phrases, you cannot have phrases in the same pool that would have different sounds or interpretation settings for that matter.

Phrase Pools can be locked, which protects them against accidental modification.

## **Editing Libraries**

The folders, pools and phrases in a library can be easily reorganized with drag & drop. You can move or copy all items between multiple open libraries.

## **Library Sounds**

With Synfire Pro, a library can have its own rack of sounds, just like an arrangement. This allows you to collect phrases along with their originally intended sounds. When you drop such a phrase onto a track, Synfire will ask you whether you also want to copy the original rack module to the arrangement rack. This greatly simplifies the task of building an arrangement from scratch.

## **Related information**

Embedded Library (on page 23) Library App (on page 149) Import SMF (on page 264) Phrases (on page 17)

# Factories

## Generating random parameters and phrases based on musically meaningful rules

## Video Tutorial

A **Factory** is a rules-based algorithm that randomly generates new **Parameters** and **Phrases**. You can think of a factory as a kind of pre-wired modular synthesizer.

On the **Factory** tab in a **Library** you can select a built-in Factory to start a new pool of phrases from scratch. Each factory has a distinct user interface that allows you to influence its output. A generated phrase retains the settings used to generate it, which is indicated by a small factory icon.

Synfire ships with a collection of basic factories suitable for many purposes. These built-in factories are powerful algorithms that you can customize to create new factories for the style and purpose you want. They are merely a starting point for you to grow your own personal collection of phrases and parameters over time. The range of musical expressions a single algorithm can generate is vast and it takes some time to make yourself familiar with its settings.

## Important:

Every generated phrase retains the settings used to generate it, so you can return to it later and continue to spawn off new variations. A carefully configured generated phrase therefore is a new **Factory** in its own right. Instead of starting with default settings from scratch, you can select an already generated phrase to recall its original factory settings and continue making new phrases based on that.

## Note:

Factories are written in the rules-based AI programming language KIM, which was specifically designed and implemented by Cognitone for the purpose of generating musically plausible and natural sounding phrases.

# Note:

This feature is available with the **Pro** edition.

<u>Factories (on page 29)</u> are a powerful modular system for generating parameters and phrases. You should <u>make</u> yourself familiar with the concept (on page 29) before you begin.

## **Factory Types**

There are factories for Parameters, Lines and Phrases.



#### **Parameter Factories**

These factories generate basic parameter data for Velocity, Flow, Pause, Step, Skip, Harmony and others.

**Use as Module**: Their main purpose is to provide input to other factories, where you pick them from a menu and configure their settings in order to achieve a desired effect. This is similar to a modular synthesizer, where, for example, you would setup a LFO generator and choose it as input to some other module.

**Use Standalone**: Add a single parameter to a **Phrase Pool**, the phrases of which you can drop anywhere you find it useful. Since the data is musically meaningful (i.e. not just random noise), this makes a lot of sense.

## **Line Factories**

These factories generate a Figure parameter of a specific symbol type. There are line factories for Bass, Chords, Arpeggio, Melody, Question & Answer, Left Hand, Right Hand (of a piano phrase) and more.

**Use as Module**: Pick a factory from a menu and configure its settings in order to add one or more lines to another phrase factory.

**Use Standalone**: Add a single **Figure** to a **Phrase Pool** the phrases of which you can use anywhere you want.

#### **Phrase Factories**

These factories generate more complex and structured phrases by merging the outputs of one or more line factories into a mixed-type **Figure**. There are factories for **Counterpoint** that generate multiple related melodic lines and factories for **Piano** that generate phrases with lines for bass, chords and melody combined.

## **Line Factory Algorithm**

Most line factories generate a **Figure** using the following algorithm. There are other algorithms and there may come additional algorithms in the future.



- 1. Spans: Divide the phrase in to spans and assign segment types to them.
- 2. Step: Generate a Step parameter with another factory.
- 3. Pause: Generate a Pause parameter with another factory.
- Segments: Fill the Figure from left to right with symbols moving up and down according to each segment type, while skipping any steps that fall into a pause.
- 5. Apply other parameters like Velocity and Flow.

Every parameter can be generated by another factory or provided by an <u>outlet (on page 45)</u>. This way you obtain control over the results of a line factory to a great extent.

## **How To Use Factories**

Despite the somewhat specific name, a **Piano Factory** can be used for a much wider range of instruments and musical styles. Since it generates lines for bass, chords and melodies from a single algorithm, you basically obtain multiple closely related elements that greatly work together. You can copy either of its lines to any instrument. Most piano phrases make great EDM patterns when you copy their lines to different electronic instruments.

Likewise, the output of a **Counterpoint Factory** is not limited to organs or piano. Many counterpoint phrases sound fantastic when distributed over multiple instruments, or played with a guitar.

**Related information** 

Factory Tab (on page 127)

# Sketches

## Conducting music in real-time

A **Sketch** renders a short passage of music in real-time. You can play Sketches on **Palettes** and <u>Progression (on page</u> <u>14</u>) editors as an alternative for the default chords-only playback.

Enable or disable a Sketch from the *Playback* menu.

You can create a Sketch with File > New > Sketch or by using the Make Sketch command of the Container menu.

## Limitations

Since a Sketch is rendered in real-time, only a few milliseconds ahead of a new harmony you ask it to play, it has no opportunity to properly prepare for a chord change in advance. This can lead to artifacts like pitch leaps and truncated notes. Depending on musical style, this can be more or less noticeable.

Due to the computing power required for real-time rendering, the number of instruments in a Sketch is limited.

## **Related information**

Sketch App (on page 147) Making a Sketch (on page 263)

# Audio System

## About instruments, devices, racks, Drones and the general architecture of the Audio/MIDI system

Synfire separates its user interface from the **Audio Engine**, which is running in the background and hosting all audio plug-ins. Both are separate processes that communicate with each other over the local area network. This allows for multiple engines to be run on different computers in your LAN.



Figure 4. Parts controlled by are orange and other parts are blue.

## i Tip:

When a DAW is synchronized in order to run alongside Synfire, a third program comes into play. Therefore, in order to keep things simple, we recommend you begin without adding a DAW to the mix until you feel comfortable with the novel work flows of Synfire. In contrast to juggling the complexity of multiple programs at the same time, the <u>Audio Engine (on page 41)</u> is managed automatically in the background avoiding distraction while you work on your music.

## Note:

Audio output of <u>Audio Engines (on page 41)</u> is not currently sent back to and mixed in Synfire. If you run multiple engines on different computers in a LAN, audio output of each engine is sent to the audio interfaces of each computer individually.

## Instruments

## Instruments interpret phrases to render MIDI output.

An **Instrument** interprets the <u>Parameters (on page 2)</u> of a <u>Phrase (on page 17)</u> to render MIDI output, taking care of its <u>Playing Ranges (on page 35)</u>, <u>Category (on page 34)</u> and other properties. Consequently, if you alter the properties of an instrument, this will almost certainly alter MIDI output.

The properties of an Instrument include

- Name
- Sound (on page 37)
- <u>Category (on page 34)</u> (or type)
- Playing Ranges (on page 35)

- <u>Custom Controllers (on page 199)</u> and <u>Articulations (on page 201)</u>
- Program/Patch selection details

Instruments show up in an <u>Arrangement (on page 19)</u> as labeled <u>Tracks (on page 20)</u> or as strips on a Matrix Console. There are also <u>Global Instruments (on page 40)</u> (and invisible instruments) at work, when you surf a <u>Palette (on page 26)</u>, preview a <u>Progression (on page 14)</u> or browse a <u>Library (on page 28)</u>.

## Note:

Instruments are used over the course of an entire arrangement. Changing the sound midway is not supported. Create a new instrument for every sound you want to use.

## **Multiple Voices**

Multiple tracks may use the same **Sound** for the purpose of separating voices. Technically, these are separate Instruments, although musically you would consider them separate voices. For example, you can add two instruments for the left and right hand of a piano part, or four instruments for the individual voices of a counterpoint part. These instruments will use different **Playing Ranges**, <u>Interpretation (on page 301)</u> settings and other parameters, so you can control the rendering of each individual voice.

## Category

Synfire organizes <u>Sounds (on page 37)</u> in a hierarchy of categories that denote the type of instrument they are resembling. Sound categorization has clear advantages.

- 1. Browsing large collections of sounds is easier.
- 2. Categories provide default **Playing Ranges** (on page 35), so if you assign a category to a sound or instrument, it will assume its playing ranges, unless specific ranges are already assigned.
- 3. Synfire can automatically find a replacement if a sound is not currently available. This happens, for example, when a file was created by another user or in a different studio.
- 4. Sounds can be previewed with a custom Phrase that was assigned to their category.

For synthetic or abstract sounds, it is sometimes not obvious which category to assign. In that case you should assign a category that comes close to the **role** a sound would take on in an arrangement. For example, if a synthetic sound is great in the lower pitch range, you may want to assign it the category **Synth Bass**.

## Inheritance

An <u>Instrument (on page 33)</u> uses the category of the <u>Sound (on page 37)</u> currently assigned to it (default), until you override this setting in an <u>Arrangement (on page 19)</u> or anywhere else an instrument can be configured.

## Note:

The category tree of Synfire cannot replace the sophisticated sound matching and browsing schemes many plug-ins have to offer. The main purpose of a category is to find replacements for a missing sound and provide Synfire with a hint how a sound is expected to be played.

## **Playing Ranges**

## Setting the target MIDI range for an instrument

Almost all <u>Parameters (on page 2)</u> that make up a <u>Phrase (on page 17)</u> are relative (parametric) and thus require context to be resolved against. There is no way to tell in advance what the final MIDI output will be, until all of an arrangement's container hierarchy and <u>Harmony (on page 300)</u> is considered.

In contrast to a piano roll in a DAW, looking at a Figure (on page 2) in Synfire does not tell you anything about the MIDI pitches that will be used. So for Synfire to be able to dynamically transpose and transform musical expressions in order to make them land within a desired pitch range, three playing ranges are set for every Instrument: Lower, Middle and Upper.



The total of all three ranges combined implies the maximum pitch range an instrument is capable of playing. Natural instruments have physical limits and even synthesized sounds might not sound great across the entire spectrum of MIDI notes. You decide what the best playing ranges are for any particular instrument.

Each range has an adjustable Typical Pitch that works best for the instrument in the respective range.

- 1. Lower: Typical Pitch should be optimal for using the instrument in a bass voice. The magenta bass symbols of a Figure (on page 294) use this range by default.
- Middle: Used for chords and melodic segments by default. Set the Typical Pitch to what sounds best for narrowly voiced chords.
- Upper: Not used unless you assign it to a Figure (on page 294) segment. Set the Typical Pitch to what sounds best for a bright lead melody.

## i Tip:

Ranges can be set in advance by creating a **Device Description** (*on page 36*) that aggregates this and other meta-data for all sounds it provides. Or you do it on-the-fly while an arrangement is evolving: Fire and forget. The former is more sustainable in the long term, while the latter gets you going faster on an ad-hoc basis.

## i Tip:

Each range should be at least one octave wide. For instruments with a narrow total pitch range (e.g. flutes), you can set all three ranges to the physical limits of the instrument and have the **Typical Pitch** merely reflect slightly different registers.

## 

Do not confuse these ranges with voices! For counterpoint movements with multiple voices, you would set the ranges for each instrument that renders one of the voices.

## Inheritance

<u>Instruments (on page 33)</u> use the ranges of the <u>Sound (on page 37)</u> that is currently assigned to them (default) until you override them in an <u>Arrangement (on page 19)</u> or anywhere else an instrument can be configured (e.g. for <u>Global Instruments (on page 40)</u>).

## **Scanning Ranges Automatically**

For sounds of any plug-in, you can let Synfire scan the ranges of a sound automatically. Press the Scan button and it will analyze the acoustic properties of the sound to determine its ranges. For this to work, Synfire needs to know the sound's category.

#### **Editing Ranges**

Use the Playing Ranges Inspector (on page 57) to adjust the ranges of an instrument or sound manually.

#### **Related information**

Editing Playing Ranges (on page 57)

## **Devices**

## Device Descriptions keep important meta-data for sounds

All internal sound generators, audio plug-ins (instruments and effects) and external MIDI hardware (synthesizers) are referred to as **Devices**. The term is also used as a short form of **Device Description**, because that is basically the only thing you deal with in order to set up devices for use with Synfire.

#### **Device Description**

A **Device Description** is a small file that contains meta-data about an actual device (synthesizer, plug-in, sampler preset), the <u>Sounds (on page 37)</u> it has to offer and how these are selected. In particular this meta-data includes for each sound:

- Instrument <u>Category (on page 34)</u>
- Playing Ranges (on page 35)

- Custom Controllers (on page 199)
- Custom Articulations (on page 201)
- Sound bank and program (patch) selection commands

Although all of the above can be set anew for every instrument in an arrangement as you go, we strongly recommend you create a **Device Description** for those sound libraries and synthesizers you will be using most frequently. Especially if **Custom Controllers** and/or **Custom Articulations** are involved.

## Sounds

The term **Sound** refers to a patch, program, or preset provided by a <u>**Device** (on page 36)</u>. For an <u>**Instrument** (on page 33)</u> to be able to render music, it needs to be assigned a **Sound**. This is referred to as <u>**Sound Assignment**</u> (on page 232)</u>.

A Sound cannot be physically accessed in Synfire. It is merely an address used to find and select a real sound connected via MIDI or hosted as audio plug-in, and then send MIDI data to it.

## Sound vs. Instrument

While a Sound is a permanent part of a **Device** (on page 36), an Instrument lives in an **Arrangement** (on page 19), determining how to play a Sound. Multiple Instruments with different properties may use the very same Sound, such as in a counterpoint part where multiple voices move along in different ranges.

## Related information

Assigning Sounds (on page 232)

## **Device Templates**

**Device Templates** provide complete meta-data for **Sounds** that can be copied to other **Device Descriptions** or **Instruments** when you need them.

## Racks

## Hosting your sounds.

A **Rack** holds one or more **Rack Modules**. Such a module may host an audio plug-in, the **Embedded GM Synthesizer**, a **Drone** in a DAW, or a **MIDI** port that connects to some external hardware.

There are three types of racks.

- 1. The **Global Rack** is hosting sounds that are available to the entire system and all open files.
- 2. The **Arrangement Rack** that is included with every <u>Arrangement (on page 19)</u> to host sounds for the arrangement's instruments.
- The Library Rack that is included with every Library to host sounds required by its phrases in order to best demonstrate their character and usage.

Racks are loaded automatically with their respective files. The only rack you would open manually is the **Global Rack**. When a rack is unloaded, all its audio plug-ins are also unloaded from memory.

## **Related information**

Global Rack (on page 39) Arrangement Rack (on page 24) Rack Editor (on page 129)

## **Rack Modules**

A **Rack Module** somewhat resembles the look of a 19-inch rack-mount device and takes on the role of managing the audio side of one or more instruments. It is not unlike the channel strip of a mixing console.

Since a MIDI port has a capacity of 16 channels, every Rack Module can supply up to 16 different instruments with sounds. Provided the loaded plug-in or connected external hardware is capable of receiving on that many channels, of course.

## **Rack Module Preset**

## Save time with presets for your most frequently used sounds

A **Rack Module Preset** includes the current state of a plug-in, plus a copy of the **Device Description** (on page 36) it is associated with. It can be used to restore both in an instant, simply by dropping it on a rack module or track.



Save a new preset from the plug-in button of a <u>Rack Module (on page 131)</u> to add it to the <u>Rack</u> <u>Module Library (on page 47)</u>. It will be listed in the browser on the sidebar automatically, where you can rename, move and edit it. But most importantly, you can grab the preset in the browser and drop it on any **Rack**, **Rack Module** or **Instrument** to restore it.

## Note:

Synfire also uses rack module presets to automatically detect the device to be used for a plug-in by comparing the current plug-in state with those that have been saved as presets.

## **Related information**

Sidebar: Rack Module Library (on page 47) Rack Module (on page 131)

## **Transient Rack Modules**

## Large sound collections on-the-fly.

Rack modules Synfire happens to find on an **Audio Engine** that are not part of any particular arrangement, are transient (temporary). The sounds of transient modules however are available for use, like any other module's.

If you open a project in your DAW that was saved with Drones not belonging to any particular arrangement, these Drones will appear on your racks as transient modules. Provided these modules have once been saved as a preset, Synfire will automatically detect the **Device Description** to use when it appears online again.

When the DAW goes offline (the project is closed), the Transient Drones simply disappear from all racks again.

## Notice:

This is an experimental feature that has proved to be useful for some users with very large sound libraries. You should not rely on it when permanently archiving your projects. If Cognitone ever removes this, we will provide a means to migrate these modules back into regular racks.

## Special Rack Modules

## AUX

For each **Audio Engine** you can add one AUX module to the **Global Rack**. It receives the signals from all **AUX Send** knobs of other modules, processes the mix and sends its own output to the MASTER module.

It is useful for use with a reverb or similar effects processor.

## MASTER

For each **Audio Engine** you can add one MASTER module to the **Global Rack**. It mixes the output of all other modules. The **Gain** knob controls overall output volume. On the **Matrix Console** a slider is mapped to this knob.

## *i* Tip:

You might want to load a multi-band compressor as **Insert Effect** to balance stark volume differences while you compose.

## **Global Rack**

## Shared sounds for palette surfing, progression preview and arrangements

The sounds provided by a **Global Rack** are loaded only once and remain available for any projects you subsequently open. Much like a real hardware rack in a studio that provides sounds for multiple songs you might want to produce. This is especially handy when you open multiple related arrangements at the same time, while you edit and play them in any order and transfer phrases with drag and drop.

Most importantly, the global rack provides six <u>Global Instruments (on page 40)</u> with sounds that are used to play <u>Palettes (on page 26)</u> and preview <u>Progressions (on page 14)</u> and <u>Phrases (on page 17)</u>, including the metronome and other audible feedback.

Maintaining multiple global racks has many benefits.

- Switch between musical styles.
- Compose a number of related arrangements based on the same sounds (e.g. a film soundtrack).
- Experiment with different versions of an <u>Arrangement (on page 19)</u> without the need to reload sounds.

The rack is a file you can open, save or close like other documents. You can create as many global racks you want and use them in association with <u>Arrangements (on page 19)</u> or <u>Libraries (on page 28)</u> as needed. However, only a single global rack can be open at a time.

## **Organizing Global Racks**

The current global rack is saved automatically when Synfire is quit, unless you discard the changes. If you work with multiple global racks, save them to the Config/Racks folder with *File > Save As ...*.

## Associating a Global Rack With an Arrangement

Synfire associates the current global rack with a new arrangement by default. That is, you will be asked whether to load the rack when you open the arrangement. If you associate multiple arrangements with a specific global rack, save it in the same folder next to the arrangement with *File > Save As ...*.

#### **Related information**

Audio/MIDI Setup App (on page 156) Racks (on page 37)

## **Global Instruments**

In order to be able to play a **Palette**, preview the phrases in a **Library**, or listen to a chord **Progression** (on page 14), Synfire needs a basic selection of sounds to be always available, whether an arrangement is currently open or not. This is accomplished by the **Global Instruments**.

#### Piano

Standard instrument used for all kind of audible responses. Synfire uses it for previewing scales and chords in the <u>Catalog (on page 16)</u>, or rendering melodies when you work with the <u>Harmonizer (on page 15)</u>. Although you can choose any sound, a piano is best suited for this general purpose.

#### Chords

Used for sustained chords, e.g. when you surf a **<u>Palette** (on page 26)</u>. This should be a lush string section or a warm synthesizer pad.

#### Bass

Used for the bass voice in **Chords Only Mode**, the <u>Harmonizer (on page 15)</u> and the <u>Palette (on page 26)</u>.

#### Guitar

Only used in specific factory preset Sketches (on page 32).

#### **Drums**

Used for drums and percussion, when no other sound is available.

#### Metronome

Used as a count-in for recording. Pick a sound you want to use for the metronome clicks. It should have a low latency.

For every global instrument you assign a <u>Sound (on page 37)</u>, set a default volume and pan and edit its <u>Playing</u> <u>Ranges (on page 35)</u> as needed.

## *i* Tip:

Since the sounds of these instruments are hosted by the **<u>Global Rack</u>** (on page 39), they get replaced when you load a different Global Rack. It therefore makes sense to create your global racks based on a minimal set of sounds that always work for you.

#### **Related information**

Instruments (on page 33)

## Audio Engine

## Background application that hosts audio plug-ins on behalf of Synfire

The **Audio Engine** is a background application running in the background of the Synfire user interface application. It is hosting audio plug-ins and effects on behalf of Synfire.

## **MIDI Routing**

About the signal flow of MIDI data

## **Input And MIDI Through**

On the **Inputs** tab, you can enable the **Input** switch for individual MIDI ports and Audio Engine ports (*Playback > Audio/ MIDI Setup*).

These ports collect and merge all incoming MIDI data for recording and playback. If **MIDI Through** is enabled, the incoming data is forwarded to the **Instrument** you selected last. If you want incoming data to be sent to a different instrument, just click on its name on the **Track Sheet** (*on page 98*) or in the list of **Global Instruments** (*on page 171*).

## **Targeting Specific MIDI Channels**

It is usually not necessary to micro-manage MIDI channels, as Synfire uses the <u>Device Description (on page 36)</u> you selected for a <u>Rack Module (on page 38)</u> to manage them automatically. That is, if you select a particular sound for an instrument, the device description already knows where to send MIDI output.

If you need to address a specific MIDI channel, open the <u>Sound Wizard (on page 193)</u> for an instrument and create a new rack module on the desired MIDI port. On page two, you pick the **Fixed Channels** option and on the last page you pick a channel, name and category for the sound. Then the wizard creates and saves a device description for you.

You can also add a specific channel to an existing **Device Description** (on page 36). On the device description editor, go to the **Variants, Channels** tab and add the desired channel. On the right sidebar you edit the properties of the sound found on that channel.

## **Dynamic Sound Allocation**

Unlike a DAW, Synfire can open multiple arrangements and libraries at the same time, which may share the same devices on the global rack.

If Song A uses 12 sounds on **Soundcase** and Song B uses 10 different sounds on **Soundcase**, they can still both be open and play, although there are only 16 channels available. The song that comes to the foreground claims and selects the sounds it wants, replacing any sounds that the previously active song may have claimed and vice versa.

Similarly, within a single arrangement if an instrument needs a new sound on **Soundcase** (or any other multi-timbral instrument) it looks for the next unused channel and claims it. If an instrument is removed, its channel becomes available again. You don't have to micro-manage that.

## Note:

This only works for devices that support program (patch) selection via MIDI program change messages. For virtual instruments that require you to browse and select sounds using their own editor window, you must use the **Sound Wizard** (on page 193) to set them up and set **Playing Ranges** (on page 35) and **Category** (on page 34) by hand.

## **Related information**

Assigning Sounds (on page 232) Devices (on page 36) Instruments (on page 33) Live Chord Detection (on page 262) Rack Modules (on page 38) Recording (on page 260) Sounds (on page 37)

## **Cognitone Transport**

## ReWire module for synchronizing the transport of a DAW with Synfire

Since a Drone is merely a plug-in, it is unable to reliably control the transport of a DAW (tempo, locate, start, stop, pause). This is handled by the **Transport** ReWire<sup>™</sup> Device installed together with Synfire. The ReWire<sup>™</sup> device is loaded automatically by a DAW and communicates with Synfire over the local network. This is managed transparently in the background.

Some DAW require ReWire<sup>™</sup> to be enabled before use. Please check the documentation of your DAW.

## Drones

## Plug-ins that operate in a DAW on behalf of Synfire

Synfire is able to synchronize with and remote-control a DAW using **Drone** plug-ins. The Drones are hosting guest plug-ins on behalf of Synfire to ensure 100% accurate timing during playback, when Synfire is feeding the Drones with MIDI data over the network slightly ahead of time. This synchronization allows for already recorded audio tracks (e.g. vocals, guitars, natural instruments) to run alongside the music you are composing while it is evolving.

## Troubleshooting:

Drones must be fully loaded in a DAW to connect with Synfire. Some DAW suspend or delay loading plug-ins in order to speed up the loading process. This makes an associated arrangement come up broken when you open it. Check the project settings of your DAW and make sure all plug-ins are always loaded, whether or not the DAW thinks they are currently needed.

Ø View ▼ Show Channel Strip ▼ Show Insert ▼		🖀 Inst 2	$\Box$
Bypass Compare -			<ul> <li>Copy Paste</li> </ul>
02 Kore Player AudioUnit *	Abyss Flowers Private: Audio Fun	Loc	al Export <mark>Online</mark> Load Editor
		Drone	

Be aware though that Synfire has only very limited control over a DAW besides setting the tempo and moving the transport. As much as we wished otherwise, there is little in the way of accessing track data of a DAW from inside a plug-in. DAWs are just not prepared for this and any plug-in loaded into a DAW is completely at the mercy of its host.

## Note:

Always save a Synfire arrangement *before* you save the associated project file in your DAW. This is because Synfire transmits all MIDI data and important meta data to the Drones to have them saved with the DAW file.

## i Tip:

While it is possible to build a **Global Rack** based on Drones, we do not recommend it. The **Audio Engine** is better suited for that. It also offers a lower latency when playing live from your keyboard hardware.

## **MIDI Drones**

The **MIDI Drone** is a light-weight version of the **Drone** that loads into your DAW as a **MIDI effects plug-in** to feed it with a stream of MIDI data that is rendered by Synfire. It cannot load a guest plug-in.

Be sure to route a MIDI Drone's output to the desired destination inside the DAW, e.g. an internal instrument. Depending on the capabilities of your DAW, it may be possible to chain multiple MIDI effects in a row.



Not all DAW support MIDI effects plug-ins in the same way. There may be restrictions as to where such a plugin can be loaded, if at all. Please consult the documentation of the DAW.

## **Related information**

Synchronizing With a DAW (on page 238)

# **User Interface Basics**

## Fundamental principles you should know to get started more quickly

A few things the user interface of Synfire is based on and that are good to know before you start.

## Mouse

Knowing some basic gestures greatly improves your user experience and productivity.

## **Right-Click**

The secondary mouse button (right-click) offers convenient access to may operations and contextual menus. Synfire makes extensive use of this. Point the mouse at any object and open a menu with options and commands with a right-click on it.

## i Tip:

You can explore many available actions by right-clicking on an object.

For whatever design reasons, Apple discourages the use of the secondary mouse button. It can be enabled in **System Preferences**. We strongly recommend you do so when working with Synfire. Alternatively you can simulate a right-click by holding down the **Control** key while you click on an object.

## **Double-Click**

Throughout the user interface of Synfire, a double-click on something often means to look deeper into its details or get out of a detailed view again.

For example, a double-click on a **Parameter View** in an arrangement opens the more detailed **Phrase Editor**. Another double-click into the slack area of its **Parameter View** jumps you back to the arrange view. The same holds true for **Containers** in the **Structure View** and their counterparts on the **Overview** page.

A double-click is also available for executing a command from the toolbar. For example, a double-click on a note length quantizes the current selection to that grid. A double-click on a **Figure Symbol** type applies that type to the current selection, and so forth.

## Parameter Outlet

## A hub for drag & drop or copy & paste of parameter data

**Parameter Outlets** indicate the presence of parameter data. They appear in groups (blocks), or as an isolated outlet for a single parameter. They allow for conveniently<u>copying or moving parameter data (on page 242)</u>.



You can scroll parameter outlet blocks with the mouse wheel and collapse individual groups to make room for others.

## References

Some **Parameter Outlet** appears standing alone, only for a single parameter. This is a reference (or shortcut) to some important parameter, e.g. the <u>Harmony (on page 300)</u> of the current container, or the <u>Preview (on page 307)</u> progression to use for phrase playback by default.

## Colors

The LED of an outlet indicates availability of parameter data.

Figure

A brightly lit LED indicates physical data is present.

#### Harmony

A dimmed LED indicates the parameter is assigned somewhere up in a parent <u>container (on page 21)</u>, or a global parameter or system default is used. You can make a **Snapshot** of these, which puts a physical copy in place that you can edit.

Velocity

A blue LED can be dragged off even though there is no physical data. It will be extracted from the Figure (on page 294) when there is one.

Preview

A yellow LED indicates a parameter shared by all phrases in a phrase pool.

## **Multiple Selection**

Multiple objects can be selected using the following tools:

## £

Drag open a span. The objects inside the span will be selected.

## []

Click a context (chord) and hold down the 🕱 Control key while you click on additional contexts.

## 8

Drag a rectangle around multiple objects, or hold the **# Control** key while you click on additional objects.

## ഷ്ഠ

Drag a rectangle around multiple symbols, or hold the **#** Control key while you click on additional symbols.

## **Crash Reports**

Synfire is designed to be robust against minor bugs. In many cases it can continue to be used after an error has occurred. Just dismiss the error message with **Continue** where it is offered as an option. However, as a precaution you should save your work under another file name when such an error occurs repeatedly.

All crash reports are collected so you can later submit them with *Help > Online Updates* to our servers for review. We encourage you to do so as it helps us greatly with improving the software and ironing out bugs.

# Chapter 2. User Interface

## About the user interface of all apps, parts, inspectors, panels and more

The topics under this section are all entry points for the embedded **Help Browser** where they show up when you hover over or select objects in the application. We therefore suggest you open Synfire and do just that, instead of reading these sections like chapters in a user manual.

## **Related information**

Help Browser (on page 85)

# Sidebar: Rack Module Library

Adding sounds made easy

## Video Tutorial

This browser on the left sidebar shows various items that can be used to create or change a **Rack Module** in order to provide sounds for instruments. Open the browser with **O** on the window toolbar.

Browse and search for items that you can drop on an **Instrument**, a **Rack** or a **Rack Module** in order to create or change a rack module and use its sounds for instruments.

Rack Module Library					
Search 🛞 [	$\overline{}$				
👾 Embedded GM Synth					
🔻 🖻 Rack Module Presets					
▶					
🕨 🖻 Kontakt					
🕨 🖻 Orchestral					
🔻 🚞 Physical Modeling					
🗅 Clarinet (AU)					
🗋 Lounge Lizard EP-3					
🗅 Lounge Lizard EP-4					
🗅 Oboe (AU)					
🗅 Soprano Sax					
🗅 String Studio VS-2 (AU)					
🗅 Strum GS-2 (AU)					
🗅 Tenor Sax					
Samplers					
Synthesizers					
▶ Audio Plug-ins					
DAW Drones (Audio)					
DAW Drones (MIDI)					
<ul> <li>External MIDI</li> </ul>					
IAC Driver Bus 1					
IAC Driver Bus 2					
IAC Driver Bus 3					
IAC Driver Bus 4					
Preset Properties:					
Soprano Sax					
SWAM Soprano Sax (AU)					
Soprano Sax					

## $\square$

**Embedded GM Synth**: Drag it to an **Instrument** or **Rack** if you want to use a sound of the embedded synthesizer. The advantage being that your arrangement or Sketch will work on any installation of Synfire, not depending on specific plug-ins. This item is grayed out if the embedded synthesizer is disabled.

**Rack Module Presets**: All presets you have saved so far will appear here for browsing. Create additional folders and organize your presets as you see fit. In the panel below, you can edit names and comments.

Drag to an Instrument or Rack to use the sounds of this preset.

Drag a preset with an effects plug-in to a Rack Module to load the plug-in as the module's insert effect.

## Ϋ́

**Audio Plug-Ins**: Find all scanned AudioUnits, VST and VST3 listed here. A matching device description will be looked for and suggested when you drop a plug-in on an **Instrument** or **Rack**. Dropping an effects plug-in on a **Rack Module** will load it as the module's insert effect.

Ŕ

**DAW Drones (Audio, MIDI)**: If the DAW is open and has **Drones** loaded, these will be listed here. **Drag** one to an **Instrument** or **Rack** to claim it for your arrangement and populate it with a plug-in and device description.

## $\bigcirc$

**External MIDI**: Find all MIDI ports know to the system listed here. Drag one to the empty area of a **Rack**, or to an **Instrument**, if you want to send rendered MIDI output to this external hardware or software.

A port is disabled when it is already in use. You can edit the device description using the **Parameter Inspector** on the **Sound** tab to add more channels to the device.

## ••

**Global Rack Modules**: Find all rack modules of the current **Global Rack** here. Drag one to an **Instrument** or into a **Rack** to use a copy of it in your arrangement.

#### ŝ

**Global Instruments**: Drag one to an **Instrument** to use it in your arrangement. The advantage being that your arrangement or Sketch will work on any installation of Synfire, not depending on specific plug-ins.

## t+

**Global Mix**: These are placeholders for a global MASTER and AUX rack module. Drag one to the empty area of your **Global Rack** to add it to the final mix.

## **Reclaiming a Drone**

When you open a DAW project with **Drones** loaded, these will connect to Synfire and appear listed in the rack module library. Those that are still available can be used by your arrangement, but some Drones may be disabled (grayed out) because they are already in use or belong to a different arrangement.

If a Drone belongs to an arrangement that you no longer have access to, or that happens to be broken for other reasons, you have the option reclaim the Drone and add it to the current arrangement. Select it in the browser and click on **Lock** icon to unlock it and add it to your arrangement.

## **DANGER**:

You should do this only to repair a broken setup, or when the link between DAW project and arrangement somehow got lost. Once a Drone has been reclaimed, the original arrangement that owned the Drone will no longer find it.

## **Related information**

Rack Module Preset (on page 38) Rack Module (on page 131)

# Sidebar: Phrase Library

## Your repository for parameters, phrases and ideas

This part on the left sidebar lists all **Phrase Pools** in the currently selected **Library**. You can select the arrangement's own **Embedded Library** (*on page 23*), or any external library file from the drop-down menu or the main Library menu.

Open the browser with the book icon on the window toolbar.

Browse and search for <u>Phrase Pools (on page 28)</u>, <u>Phrases (on page 17)</u> and <u>Parameters (on page 2)</u> that you can drop on a **Track**, <u>Parameter View (on page 144)</u>, a <u>Parameter Outlet (on page 45)</u> or anywhere a phrase or parameter can be used.

## Synfire User Manual | 2 - User Interface | 50



## **Adding New Phrases**

Apart from dropping phrases, parameters, tracks and containers, you can add new parameters and phrases to the current library with **E Control E** or *Library > New Phrase From Selection* anywhere in the arrangement where you have selected a Figure or other parameter with the **Span** tool.

- If you selected a span on the Figure parameter, the new phrase will also include all other parameters that typically belong to a phrase.
- If you selected a Take, only Take and Harmony will be collected.
- Any other selected parameter will be collected only as a single parameter.
- If needed, you can also drag any parameter to the library with the mouse.

This is very convenient when you harvest an imported file, or encounter a parameter that you find useful for other purposes. It basically works with any parameter. When you grab a **Figure** this way, Synfire also copies the currently associated **Harmony**, **Tempo**, **Interpretation** and other parameters in order to get you a complete reusable phrase.

## Note:

The phrase library sidebar must be visible for the command to be enabled.

## Toolbar

## **Library Selection**

In addition to the embedded library, you can select <u>library files (on page 28)</u> from this menu to open and browse. Files not listed here can be opened with *Library > Open* ... from the main menu.

The three most recently used <u>library files (on page 28)</u> are kept open in the background, so you can quickly switch between them without the need to reload them from disk. You will be asked whether to save pending changes when the least recently used library needs to be closed or when the **Arrange** window is closed.

## 

Opens a menu with options for the selected phrase pool or folder.

## **Browser**

Contents of a <u>Library (on page 28)</u> are organized in a hierarchy of folders. You can arrange folders freely with the mouse. Pools can be moved or copied between folders. Phrases can be moved or copied between pools.

## **Phrase Pool**

- If you Drag a pool to a destination that expects a phrase, this will use its default phrase as explained the section Phrases below. For pools that contain more than one phrase, you may want to pick a specific phrase instead.
- Drag a pool to a Snippet Group to use all its favorite phrases for live playback.
- Double-click on a pool to jump to the Library page where you can edit it.
- Alternatively use keyboard shortcuts Cut, Copy, Paste, Delete or Duplicate.

## Phrase

- Double-click on a phrase to insert it into your arrangement.
- Drag a phrase to an Instrument Name on the Structure page to replace the entire phrase as a whole.
- Drag a phrase to another Folder or Library to move it. Hold down 📉 Ctrl to copy it.
- Drag a phrase to a <u>Parameter View (on page 144)</u> on the **Structure** page, or to the **Phrase Editor** in order to append, replace, insert or merge the phrase.

Drag a phrase or parameter to a **Parameter View** to replace the span being highlighted under the mouse.

- Hold down 🕱 Control to insert dropped data at the target position.
- Hold down 🛈 Shift to merge dropped data with existing data.
- Drag a phrase to a Snippet to use it for live playback.
- Alternatively use keyboard shortcuts Cut, Copy, Paste, Delete or Duplicate.

## Folder

- Drop a track or instrument name into a folder to add its phrase to the library.
- Drop phrases or folders from other libraries to copy them.
- Drop a Container to add all its phrases.
- Drop a Parameter Outlet to add the current parameter data to the library.
- Move folders to reorganize the library.

## **Containers, Snippets and Memories**

Containers, snippets and memory slots can also be stored permanently in a library. Simply drag them into a library. However, their content is static. You cannot add or delete anything, but you can use individual phrases as usual.

## Container

Drag it into an arrangement or onto the snippet grid to restore it (Pro). If the container contains aliases, you may need to restore them first. You can use individual phrases and parameters inside as usual.

#### Snippet

Drag it into an arrangement or onto the snippet grid to restore it (Pro). You can use individual phrases and parameters inside as usual.

#### **Snippet Memory**

Drag it to a memory button above the snippet grid to restore it. However, original snippets that no longer exist will not be restored. In such a case, you can manually restore the snippets found inside.

Double-click on the object to schedule its content directly for playback without having to restore the memory slot. In this way, you can keep as many (virtual) memory slots in a library as you like. However, if one of the snippets used has been deleted in the meantime, it will no longer be played.

## Tabs

## (i)

Open this tab for information on the selected Phrase Pool.

Show individual phrases in the selected **Phrase Pool** (on page 28), so you can drag them to the arrangement.

Drag a phrase or parameter to a Parameter View to replace the span being highlighted under the mouse.

- Hold down **#** Control to insert dropped data at the target position.
- Hold down 👔 Shift to merge dropped data with existing data.

## ~

Shows an **<u>Outlet** (on page 45)</u> for every available parameter in the selected **<u>Phrase</u>** (on page 17) that you can drag to your arrangement.

## **Switches**

## 0

Preview phrases in the library using the **Instrument**, <u>Tempo (on page 313)</u> and <u>Harmony (on page 300)</u> in the currently selected container. This way you can assess how a phrase will sound in your arrangement.

## Note:

You may not hear the phrase if another container plays a different phrase for the instrument at the same position.

## く》

Preview phrases and parameters in the library along with all other instruments in the currently selected container. This way you can assess the effect the phrase or parameter will have on your arrangement. This works with any parameter and the **Global Parameters** track, too.

## Note:

You may not hear the phrase if another container plays a different phrase for the instrument at the same position.

## =5

Transpose the **Harmony** of the currently selected phrase to the key that dominates the current container in the arrangement (or a major/minor parallel). This transposition is also performed when you drag the phrase somewhere.

## 

Copies the current phrase into the arrangement permanently.

## ட

Phrase pools can be locked against unwanted modification. Phrases can still be viewed and KIM Factories can be run to generate more variations based on the last used settings. These new phrases are added to a separate pool.

## פצ

Assign a **Sound** to the phrase pool. Using the **Sound Wizard** that pops up, you can add sounds to the library's own **Rack**. The background color of the button indicates the current status of the sound.



- 1. Transparent: A sound on the Global Rack is used, or a sound is not currently needed.
- 2. Blue: A sound on the library's rack or the arrangement's rack is used.
- 3. Orange: The sound remembered by the pool is not currently available and a replacement is used.
- 4. Red: Neither the original sound, nor a replacement could be found.

# CAUTION:

Be careful not to copy too many unrelated phrases from other libraries into the **Embedded Library**. This would bloat your arrangement file and slow down save and load times.



## CAUTION:

You should be careful not to add too many Rack Modules to a Library, or risk it taking more time to load.

## **Related information**

Libraries (on page 28) Embedded Library (on page 23) Phrase Pool Editor (on page 123) Library App (on page 149)

# Parameter Inspector

## The multi-purpose sidebar on the right

This part shows up on several pages on the right sidebar. It can be hidden with **one the window toolbar if you need** more room for other views. Despite its name, it does not only allow for the inspection of parameters, but also provides a place for editing the selected **Instrument** and other aspects of an **Arrangement**, **Sketch** or **Library**.



## Tabs

These are the tabs showing up in the Arrange App.

~

**Parameter**: Shows a parameter-specific inspector depending on the parameter currently selected. Go here to edit the parameter.

## 夏

Instrument: Shows an inspector to edit the Sound and properties of the selected Instrument.

Container: Shows a form to edit the current container's name, length, comments and other properties.

**=**]

**Notation**: Edit **Notation Export** preferences for the instrument. Select the **Global Track** to edit general export settings for all instruments.

## $\bigcirc$

Synchronization: Edit preferences for external synchronization with a DAW or other software.

## **Parameter Commands**



## 6

Listen to an audible preview of the selected parameter. Some parameters don't support this.

## രി

Insert a physical snapshot (on page 258) of the selected parameter.

#### Delete

Remove the selected parameter from the phrase.

## $\otimes$

Insert a predefined Parameter Template from a menu.

## 

Open the *Parameter* menu for various editing options.

## **Instrument Properties**

Adjust the settings for an Instrument on the sidebar where the Parameter Inspector (on page 54) is located.

If the selected device has **Allow Modification in Arrange View** enabled, you can flip the switch **Sound Details** and edit **Sound Properties** (on page 140) directly without navigating to the **Sounds** page.



## Sound

#### Label

Label the instrument if the default (sound name) is not specific enough.

## Color

Changes the label color on the Matrix page.

#### **Status**

Indicates the status of the instrument:

- Green: Sound is available.
- Orange: Sound is a replacement, because the original could not be found.
- Red: Sound is not available.

#### **Rack Module Selection**

Select a rack module to provide a sound for the instrument.

#### **Rack Module Channel**

Select one of the MIDI channels provided by the rack module. If your desired sound is not listed, you may need to edit the rack module to add more channels and sounds.

## **Preferred Channel**

Demand from this instrument to claim a specific MIDI channel where possible.

## **Open Plug-in Editor**

Works only for plug-ins loaded by the **Audio Engine**. Plug-ins loaded by a **Drone** must be opened and edited in the DAW.

## **Custom Controllers, Articulations**

Set up optional <u>Custom Controllers (on page 199)</u> and <u>Articulations (on page 201)</u> for the instrument, overriding those of the sound and device. This is saved with the arrangement only, so the original sound and device are not altered.

## **Sound Details**

Open a <u>view on the sound's properties (on page 142</u>), which in some cases you can edit here right away.

## **Toolbar**

## ••

Jump to the Sounds (on page 128) page to edit the rack.

## Ø

Jump to the **Device Description** editor.

## 6

Preview the Sound with a phrase set for its Category.

## ß

Open the Sound Wizard (on page 193) to configure the sound interactively.

#### **Related information**

Sound Properties (on page 140)

## **Editing Playing Ranges**

Adjust the <u>Playing Ranges (on page 35)</u> of a <u>Sound (on page 37)</u> or <u>Instrument (on page 33)</u>. The Arrange App uses this vertically arranged inspector. A horizontal variant of this inspector is used everywhere else.

## Synfire User Manual | 2 - User Interface | 58

·····································		Ranges			
∞ © © ©					
Ē		≘		•=	
			_		
			=		

#### **Modify Ranges**

Drag the edges of the blue **Range** to expand or shrink the range. Or click while holding Drag the green **Typical Pitch** to best possible sound. Or click while holding the Alt key to set a particular note.

## Toolbar

## 0

Accept the default ranges of the **Sound** as specified in its **Device Description**.

## 

Scan ranges automatically by probing audio output of the plug-in. Assign a category first.

## $\bigcirc$

Learn the range by tapping on your MIDI keyboard the lowest and the highest note.

## 60

Play the preview phrase of the associated Category.

## .Ξ

Lower Range. Used for bass symbols by default.

Middle Range. Used for chord symbols and both scale symbols by default.

•=

•Ξ

Upper Range. Used only where you assign it to Figure segments.

## **Options Menu**

## Cut, Copy, Paste

Use these to copy the playing ranges between any instruments or sounds.

## **Copy Ranges From Sound**

Copy the ranges from the sound to the instrument. Basically does the same as the link icon on the toolbar.

## **Copy Ranges To Sound**

Copy the current ranges of the instrument to its selected sound. This is useful when you want to permanently save the current ranges with the device description for later reuse.

## Distribute Ranges Like ...

Scale the ranges proportionally to the playing ranges of a Category.

## **Distribute Ranges Equally**

Divide the current total pitch range into three equal parts.

## **Same Extremes For All Ranges**

Make all three ranges span the total pitch range and retain their Typical Pitch.

## **Container Properties**



#### Name

A meaningful and short name. Anything that suits your purpose will do.



Assign a color to the container.

## Length

Length of the container in terms of its own time signature (Scheme).

#### Offset

Position where the container starts, in terms of the parent container's time signature (Scheme). Allows for any duration expression, e.g.  $\frac{6m+3}{4}$ . A **special syntax** (on page 325) applies.

#### Inactive

Deactivate a container to exclude it from rendering entirely. Great for A/B testing. Aliases may be activated or deactivated independently of their original.

#### **Keep Child Containers In a Row**

Forces all children into a line without gaps, no matter how you resize or reorder them. Intended to maintain a flat song structure with successive parts.

#### **Prefer Orchestral Interpretations**

If enabled, auto-detecting Interpretation (on page 301) will prefer presets meant for orchestral instruments.

#### Comments

Any hints or documentation that helps other users understand your intentions or that supports your workflow.

## Notation Export Settings

Notation export is very sensitive to these settings, so you may need to try multiple runs until you get the desired result.

## Instrument

~	Ē		Ц,	()		
Inst	rument	Prep	Glob	al		
Abbr. I	Abbr. Instrument Name:					
Piano	c		Auto			
Stave	s:	Octave:				
1	Treble ((	3) <b>'</b>	✔ 0	$\Rightarrow$		
• 2	Bass (F)	•	✔ 0	$\widehat{}$		
3	Treble (0		~ 0	$\sim$		
4	Treble (0		<b>v</b> 0	<b>~</b>		
V Slurs		Tabl	Tablature:			
Percussion		Of	Off 🗸			
Slashes						
✓ Chord Symbols						
Merge with previous Instrument Hide						

Tune these settings to get the best results for the selected instrument. They apply to all containers.

Auto

Automatically determine suggested settings from rendered notes.

#### Abbr. Instrument Name

If the name of the instrument is too long, enter a shorter name here. Otherwise, leave the field empty.

#### **Staves**

Select the number of staves (staffs) using the left radio buttons.

## Clef

Clef symbol to use for each staff.

## Octave

Transpose exported notes up or down to better fit into each staff.

## Slurs

Have Synfire determine the use of slurs based on available information.

#### Percussion

Uses special symbols for percussion.

#### **Slashes**

Uses special symbols for rhythm.

#### **Chord Symbols**

Chord symbols are attached under the staff.

#### **Tablature**

Prints a tablature corresponding to the selected string instrument.

#### **Merge with Previous Instrument**

Use this to combine multiple instruments in one exported staff. For example, left and right hand of a piano. Likewise, this applies to multiple contrapuntal voices. The combined output of all merged instruments is exported like output of a single instrument.

#### Hide

Do not export this instrument.

## **Pre-Processing**

~	受		Ę	U		
Instru	Instrument Prep			Global		
Pre-Pro	Pre-Processing					
	Status			Auto		
Quantize Notes						
Alread	Already Quantized 🗸					
V Exp	ect Triple	ets				
📄 Ехр	ect Quin	tuplets				
Stretch	Notes Sh	orter Tha	an			
1/64	~					
Close R	Close Rests Shorter Than (Legato)					
Never 🗸						
Shortest Notated Rest						
1/64	1/64 🗸					
Shorten Overlaps						
Monophonic						
Separate Voices						
2 ~						
✓ Follow Figure Segments						

These settings are applied to the rendered sequence of the instrument before any notation export processing is done. You can use them to cleanup overly complex or not yet quantized rhythm and tell Synfire how many voices you expect in polyphonic parts.

#### Auto

Estimate suggested settings from current Output.

#### **Status**

This indicator shows whether the instrument is ready for export.

- 1. Green: The pre-processed sequence is valid for notation export.
- 2. Orange: Some elements will be invisible or look odd, but export should still work.
- 3. **Red**: Pre-processing is unable to eliminate invalid note positions or durations. You need to tune the settings, or edit and manually quantize all **Figure** parameters of the instrument.

#### **Quantize Notes**

Apply this quantization to all note onsets and lengths before processing it. If Synfire believes the sequence is already quantized but you still get unexpected results, you can tune this setting manually.

#### **Expect Triplets, Quintuplets**

When enabled, seemingly odd durations of notes and rests will be considered potential tuplets and quantized as such. Disable this if you are sure there are no such tuplets.
### **Stretch Notes Shorter Than**

Notes shorter than this will be stretched to this length.

### **Close Rests Shorter Than (Legato)**

Stretch notes in order to close gaps, if the gaps are shorter than this. Use this to remove unintended rests. Any remaining rests may be filtered with **Shortest Notated Rest**.

### **Shortest Notated Rest**

Rests shorter than this will be exported as invisible skips if the file format supports it. Set this to a very long duration to make most rests invisible.

### **Shorten Overlaps**

Remove overlaps by shortening notes where necessary. Note that overlaps are always eliminated per each individual voice. This setting however is applied to the entire sequence prior to voice separation. It is best suited for monophonic parts, but can also be used to clean up a messy playing style.

### Monophonic

Eliminate polyphony altogether. This is also applied to the entire sequence prior to voice separation.

### **Separate Voices**

When Synfire runs its voice separation algorithm, it will look for this number of distinct voices. Voice separation improves the look of polyphonic phrases. Note that this is independent of the number of staves, as voices may also move across multiple staves.

#### **Follow Figure Segments**

When separating voices, keep notes together in the same voice that have been rendered by the same **Figure** segment. This helps improve consistency, but also requires that your use of segments is related to voices in the first place.

# Global

~	顚		Ę	C						
Instru	iment	Prep	Glob	al						
Export Options										
<ul> <li>✓ Glu</li> <li>✓ Ha</li> <li>Co</li> <li>✓ Ac</li> <li>✓ Be</li> </ul>	obal Key armony Ir ontainer S cidental aming	Signatu nformatio Structure s	re on e							
A4	<b>∨</b> ndscape	Paper	Size							
Emb	edded N	MIDI								
✔ Ge ✔ Pr	eneral MI ogram C	DI hanges								
Dedicat	ion:									
Synfire	e Pro 2.0	.2 Build	#7 (Wir	ndows)						
Lyricist:										
Subtitle										

These settings apply to the exported notation file as a whole. Synfire remembers your last settings.

#### **Global Key Signature**

Assumes a single key signature for the entire file. Key changes are not exported as such. All notes are interpreted in the perspective of the global key.

### **Harmony Information**

Adds additional information on chords, scales and keys to the exported file, where the format supports it. This information does not necessarily affect the visual appearance of a score.

### **Container Structure**

Adds container names to the exported file. This information is saved as text markups.

#### **Beaming**

Enable this if you want Synfire to take care of the grouping of notes. Notation programs are often much better at this, but it is worth trying both options.

### **Accidentals**

Enable this if you want Synfire to export accidentals to the file. The notation program will probably figure them out on its own, but this option helps in cases where that doesn't work.

#### **Paper Size**

LilyPond PDF output only.

### **General MIDI**

Include General MIDI compatible information for all sounds.

### **Program Changes**

Include MIDI program change commands where possible.

## **Dedication, Lyricist, and Subtitles**

Depending on the exported file format, this information may or may not be printed with the final result.

# **External Synchronization**

Make your DAW, external film projector, or other hardware or software run in sync with Synfire, such that both time lines always match during playback. This is a requirement if you are using **Drones** to host sounds directly in your DAW.

# Timing

~	爂		=	ſ	<b>(</b>
Timir	ng Tr	anspo	rt	MID	N
Render	ing Head	lroom			
Tolera	nt (80m	s)			~
Global S	SMPTE О	ffset			
Frame I	Rate				
24 FP	S	~			
Global	Pre-Roll	(Measu	res)		
None		~			
🗸 Sho	w In Rul	ers			
Externa 0	il MIDI La	atency			
V Allo	w Negat able Late	tive Loc encies	ators	5	

### **Render Headroom**

Synfire renders real-time **Sketches** slightly ahead of time, in order to give you a chance to pick the next chord from a **Palette**. Extend this time if you experience drop outs.

# **Global Pre-Roll**

Make the first bar of your arrangement start later, if your DAW can't position its play head at a negative time during the small pre-roll period that precedes every playback.

### **Global SMPTE Offset**

Time code position on the external film or tape that corresponds to the start of your arrangement.

#### **Frame Rate**

Frame rate to use for **MIDI Time Code**. Even if you are not synchronizing to film, you need to set the same frame rate at both ends.

### **External MIDI Latency**

Compensate for a delay (milliseconds) that affects all external MIDI communication. This is a global setting for all outgoing MIDI ports.

### **Allow Negative Locators**

Disable this if your DAW can't position its play head before zero. Keep this enabled if you are using the **Audio Engine** only.

### **Disable All Latencies**

Transmit all MIDI messages without latency compensation. This sends notes at the same time, even if they are for sounds with different latencies. Use this if you want to record outgoing MIDI as a way to export your work.

# Note:

The **Express** and **Pro** editions allow for more detailed timing control and support external synchronization with a DAW and other devices.

### **Transport**



Synfire installs a ReWire<sup>™</sup> device called **Transport** that allows for most DAW to be controlled remotely.

## **Enable External Sync**

Global switch, same as on the top transport toolbar.

### **Enable ReWire Transport**

Ask Synfire to use ReWire<sup>™</sup> for sending transport commands to your DAW.

### Send Tempo

Also synchronize Tempo over ReWire<sup>™</sup> (recommended).

If a warning message shows up that no DAW with a ReWire<sup>™</sup> transport can be found, your DAW either does not support this, or you need to first enable this mode in your DAW.

# MIDI

~ <b>#</b>	1	Ę	<b>(</b> )						
Timing	Transpor	t MIC							
<ul> <li>✓ Enable External Sync</li> <li>✓ Enable MIDI Sync</li> </ul>									
Output									
IAC Driver E	Bus 1		~						
Protocol									
MIDI Clock			~						
▽ Customize									
Stream	MIDI Cloo	ck (Song	Ρ( 🗸						
Locate	MIDI Cloo	ck (Song	Ρι 🗸						
Start/Stop	MIDI Cloo	ck (Song	P( 🗸						
Tempo	None		~						
Latency	0								

Synchronization via MIDI is an alternative to the ReWire<sup>™</sup> Transport. You should not use both at the same time for the same DAW, but it is perfectly possible to use MIDI synchronization with additional software or hardware.

# **Enable External Sync**

Global switch, same as on the top transport toolbar.

# **Enable MIDI Sync**

You may want to temporarily disable MIDI synchronization exclusively.

# **Output Port**

Select the MIDI port to use for sending sync messages to your DAW. A port is required, even if you choose to use OSC commands only.

### Protocol

Select a preset for the synchronization protocol that your DAW can understand:

- 1. MIDI Clock (Song Position Pointer)
- 2. MIDI Time Code (MTC)
- 3. Open Sound Control (OSC)

# **Customize**

If your DAW doesn't properly respond to transport commands, you have the option to customize each command individually, mixing different protocols in order to obtain full control.

### Latency

Amount in milliseconds to send synchronization commands ahead of time to compensate for your DAW responding late.

# Harmony Settings

When you select in a chord in the <u>Harmony (on page 300)</u> parameter, you are actually editing a <u>Harmonic Context (on page 12)</u> that includes a lot more information than just a chord.



# Chord

Pick a <u>Chord (on page 7)</u> from the menu or type its name right into the input field. You can type multiple chord names separated by spaces. There are many ways you can write a chord. Learn about the <u>syntax</u> <u>here (on page 327)</u>.

### **Extension**

Pick a chord extension (on page 7) from the menu.

#### Bass

Transpose the bass note up or down. Hold down  $\sim$  Alt for chromatic increments. Hold down  $\Re$  Ctrl to reset to automatic bass selection. The bass need not necessarily be included with the chord.

### **Function Expression**

Check this if you want to enter harmonic function expressions instead of chord names (Pro only).

### **Vertical Scale**

Pick a <u>Vertical Scale (on page 8)</u> from the menu to use for building melodies over the chord.

### **Horizontal Scale**

Pick a Horizontal Scale (on page 8) from the menu.

### Relation

Pick a <u>Relation (on page 12)</u> key from the menu. Local key signature that would best match the horizontal scale (determines color).

#### Key

Pick a global <u>Key (on page 12)</u> from the menu.

### **Estimate**

Estimates keys, scales and relations for the selected chords, assuming a common underlying key. Remember that different spans in a progression can have different keys (key changes). Hence you can select and estimate multiple spans independently. Alternatively call *Transform > Estimate Keys, Relations and Scales* from the main menu.

The result depends on your current Scale Selection Preferences (on page 192).

#### **Bass Menu**

Pick a desired interval for bass. This is an alternative to setting the bass directly.

#### **Inversion Menu**

Pick a desired chord inversion to use for <u>Auto-Chords (on page 301)</u> interpretation and the rendering of green <u>Chord Symbols (on page 295)</u>.

This may not have an effect in case <u>Interpretation (on page 301)</u> decides to follow the setting of individual <u>Figure (on page 294)</u> segments instead.

# \$Ξ

**Open Voicing**: Enable this for open chord voicing. This only affects <u>Auto-Chords (on page 301)</u> interpretation and the rendering of green <u>Chord Symbols (on page 295)</u>.

This may not have an effect in case <u>Interpretation (on page 301)</u> decides to follow the setting of individual <u>Figure (on page 294)</u> segments instead.

# $\mathcal{Q}$

Minor Seconds: Enable if you are fine with minor seconds to occur in chord voicing.

This may not have an effect in case <u>Interpretation (on page 301)</u> decides to follow the setting of individual <u>Figure (on page 294)</u> segments instead.

# 0

**Alignment**: Align the chord with the previous one. A very basic voice leading algorithm that smoothes out pitch leaps. In a long progression, you should disable this occasionally, as a sort of line break.

This may not have an effect in case <u>Interpretation (on page 301)</u> decides to follow the setting of individual <u>Figure (on page 294)</u> segments instead.

# **Override Harmony**

Always use the Preview (on page 307) parameter, even if Harmony (on page 300) is available.

# **Figure Inspector**

The **Figure Recognition Settings** button takes you to the <u>Take (on page 74)</u> parameter, where you can refine the last used preferences for **Figure Recognition**. Switching back and forth between <u>Figure (on page 294)</u> and <u>Take (on page 312)</u> is part of the <u>recording workflow (on page 260)</u>.

		≡Ĵ	©			
Figure n (d) (X)	\$ ₽		l			
			gs			
Anchor Articulation	Strong Mediu Weak	n m	Reset			
None		~				
Hold Bypass VL V Look Ahead Chromatic Muted Ducking						
Completi						
Default	~	Open	~			
Default	~	Open	~			
Default Symbol Diatonic	*	Open Mappe	<ul><li>✓</li><li>And ✓</li></ul>			
Default Symbol Diatonic Chromatic	<ul> <li>✓</li> <li>0</li> </ul>	Open Mappe	ed V			
Default Symbol Diatonic Chromatic Velocity	<ul> <li>▼</li> <li>0</li> <li>71</li> </ul>	<b>Open</b> Mappe	► ed ►			
Default Symbol Diatonic Chromatic Velocity Length	<ul> <li>▼</li> <li>0</li> <li>71</li> <li>1/8</li> </ul>	Open Mappe	► ed ►			
Default Symbol Diatonic Chromatic Velocity Length Pitch	<ul> <li>.</li> <li>.</li></ul>	<b>Open</b> Mappe	ed V			

# Segment

Properties of currently selected segments (multiple selection (on page 46) is supported).

### **Anchor Strength**

- Strong: Only chord tones are considered for the Anchor (on page 295).
- Medium: Chord tones and chord extensions are considered.
- Weak: Entire scale is allowed.

#### **Articulation**

Select a Custom Articulation (on page 201) for the segment.

### Hold

Keep <u>Harmonic Context (on page 12)</u> unchanged for the length of the segment (only affects this segment). Useful to prevent harmony from switching in the middle of a glissando, for example.

### Look Ahead

If there is a harmony change in the immediate vicinity (round 1/8), the segment will responded to it early. Makes sense where notes are pushing ahead of the beat.

## Muted

Excludes the segment or individual symbol from being rendered.

### **Complete Chord**

If the current chord has more notes than the segment has symbols, assume those extra symbols are virtually stacked on top of the segment until the chord is complete (requires at least two symbols arranged vertically in a chord segment).

#### **Inversion & Voicing**

Override for this segment the default chord inversion or voicing that was set in the **Harmony** parameter. This is useful for **Chord** and **Relative** type segments only. The **Interpretation** parameter may disable this override.

### **Bypass VL**

Disable the voice leading algorithm for this segment. Creates more harmonic tension, albeit chord changes are less pronounced.

#### Chromatic

Allows any of the 12 chromatic tones to be used, whatever the outcome. Be careful with this if you want the phrase to work also in other harmonic contexts. A harmonically safe octave can be notated with a chromatic shift of 12 or -12.

### Ducking

Temporarily excludes notes rendered by other segments while this segment is playing. Affects segments on all tracks that go to the same instrument. Optionally you can include note lengths with the covered span.

# **Symbol**

Properties of currently selected symbols (multiple selection (on page 46) is supported).

### Diatonic

Line number on the Figure staff. Interpretation depends on symbol type.

### Chromatic

Offset from the diatonic step in semitones (equivalent to traditional accidentals).

# Velocity

Velocity value of the selected symbol(s).

### Length

Length(s) of the selected symbol(s).

#### Pitch

Shows absolute pitch (only Pitch type symbols).

# Note:

Velocity and length of individual figure symbols are overridden by any <u>Velocity (on page 317)</u> and Length parameters, if those are present in the phrase, or inherited from a parent container.

# Note:

It is impossible to display absolute pitch for symbols other than the **P** type. This is because a symbol may render a multitude of different pitches when the Figure is looped.

### **Related information**

Figure (on page 294)

# **Take Settings**

The <u>Take (on page 312)</u> parameter keeps your most recently <u>recorded (on page 260)</u> or imported MIDI data ready for <u>Figure Recognition (on page 5)</u>, which will convert it to a <u>Figure (on page 294)</u>. After adjusting your settings, press <u>Apply</u> to start the recognition process. The result will be stored in the Figure parameter of the same phrase.



# **Figure Recognition**

Figure Recognition (on page 5) turns flat MIDI data in a Take into a highly structured Figure. This is a complex procedure that involves a lot of estimation. If there are enough notes, the Auto button can estimate the settings and

provide a good starting point. Although sometimes you need to experiment a bit until you find the settings that work best.

Both <u>Harmony (on page 300)</u> and <u>Playing Ranges (on page 35)</u> have a big impact on figure recognition. Be sure to re-harmonize an imported phrase if you suspect harmony does not match the take. Also, when you are recording in an arrangement, make sure you play in the current key.

# i Tip:

You can select a span in a Take and run figure recognition on it selectively.

### Preset

Pick a preset that best suits the **Take**. Don't take the label "Keyboards" literally. It can be any polyphonic instrument, actually.

### Algorithm

Select a method and strategy for recognizing figures.

- 1. Auto: Estimates an appropriate algorithm automatically.
- 2. Linear: Examines the input from left to right, detecting reoccurring patterns and characteristic movements on the way. Works best for monophonic melodic voices.
- Pattern Recognition: By looking at the take as a whole, this method identifies areas on interest and compares millions of potential segments with each other. A time-intensive but effective method that does a good job recognizing polyphonic phrases.
- 4. **Rhythm Input**: Every single recorded note is turned in to a chord or bass note. This is great if you simply want to tap-in a rhythm that is supposed to be played as chords or bass.
- 5. **Simplified Bass**: Converts the input to plain isolated bass symbols only, which works fine for many popular music styles. For contrapuntal and melodic bass voices, you should use **Linear**.
- 6. Static Pitch: Converts the take to absolute pitch symbols. This is extremely fast and good for drums and percussion, or other instruments that don't follow <u>Harmony (on page 300)</u>. You can also use this to import large MIDI files more quickly and postpone figure recognition until after the import on a track by track basis.

### **Segment Types**

Select the desired symbol types you want to generate. Some algorithms allow only a single type. The **Linear** algorithm supports multiple types only when voices are separated.

### Accidentals

Create symbols with chromatic components where they are needed to faithfully recreate the input. Disable this to get more portable phrases that are easier to edit.

#### Ranges

When enabled, this assigns a specific playing range (lower, upper) to segments if they are far off the middle range. This moves more segments towards the center, visually, but also helps a phrase translate

better to other instruments that might have very different ranges. When this is disabled however, the resulting Figure is more easily editable.

#### Resolution

Notes starting within this window are considered simultaneous. This setting merely affects the grouping of symbols into segments and does not alter the timing of your result.

#### **Number of Voices**

How many parallel voices are assumed to be in the **Take**. This is for information only unless you choose to separate them.

#### Separate

Separate voices from each other before processing them. This can much improve results when voices are sufficiently apart from each other. Where the input is an entangled mess however, it can also make things worse. If in doubt, try both options and see what works best.

### Note:

Voice separation works better on isolated phrases than entire tracks worth of imported MIDI data, which may contain multiple sections with contradicting properties.

### Transpose

Transpose the input by octaves in order to best match the target instrument's playing range.

### **Sustained**

The sound is sustained while a note remains depressed. Consecutive notes that overlap each other are meant to be independent voices. Disable this to better handle decaying or percussive sounds like guitar and piano.

### **Simplify Bass**

Place bass segments around the zero line, so they will translate to whatever is the current bass, rather than attempt to recreate the original input. Makes a phrase more portable.

### **Simplify Chords**

Generalize chords such that a phrase translates better to different Harmony.

### Apply

Start **Figure Recognition**. Depending on complexity this can take a while. A log console will open if the process turns out to be running longer than expected. Results are saved o the <u>Figure (on page 294)</u> parameter. After this you can optimize the settings and repeat figure recognition until you are satisfied with the result.

### Terminate

Prematurely end the recognition process if it takes too long, or if you want to try different settings.

# Tags

You can help **Figure Recognition** achieve better results by grouping symbols into segments or tagging their supposed types beforehand. In the **Parameter View** (*on page 144*) of **Take** (*on page 312*), simply group segments and change their symbol types as you would do with a **Figure** (*on page 294*).

# **Recording Options**

Under the tab with a red circle, you'll find several options that will alter your input before it is processed by **Figure Recognition**.

# Quantize

Select a grid to destructively snap all notes to immediately after recording.

# Monophonic

Reduce recorded input to a single voice before processing it.

# Overdub

Don't clear the phrase before recording, that is, add whatever is recorded to the existing Take.

# **Related information**

Recording (on page 260)

Figure Recognition (on page 5)

# Interpretation Settings

# Form



Form settings control the transformation and rendering of Figure (on page 294) segments.

# Anchors

How the pitch of an Anchor is influenced by the previous anchor:

- Independent: Anchors don't influence each other.
- Linked within 1/8: If the previous anchor is within this distance, an attempt is made to map the current anchor relative to that.

Only anchors of the same symbol type can influence each other, where a chord change falls between them and the resulting difference to the notated pitch is not too extreme.

### Range

How to deal with segments that happen to reach outside their assigned playing range or the total pitch range of the instrument.

- **Open**: Allows all notes outside the instrument's total pitch range to pass through. This is potentially dangerous, as it might randomly trigger key switches or have other unwanted side effects.
- **Clip**: Keep a segment as rendered and drop any notes that fall outside the total pitch range of the instrument. This is safe but may skip some notes.

- **Shift**: Transposes a segment as a whole in order to fit it into the individually assigned range (upper, middle, lower), or at least the total pitch range of the instrument. Notes that happen to fall outside the total pitch range are dropped.
- Fold: Breaks down a segment into smaller parts in order to fit it into the individually assigned range (upper, middle, lower), or at least the total pitch range of the instrument. Notes that still fall outside the total pitch range are dropped.

### **Limit Strictly**

When applying **Shift** or **Fold**, limit output to the individual playing range of a segment (upper, middle , lower). Otherwise the total pitch range of the instrument is allowed.

#### **Preserve Harmony**

When applying **Shift** or **Fold**, do so in increments of octaves rather than scale steps. This is useful where harmony is more important than relative melodic movement.

### **Strings**

Select a particular tuning for a string instrument. The tuning controls how segments of type **Chord** are rendered.

#### Auto-Split

Divide and transpose long sustained notes that create unwanted dissonances by violating the <u>Harmonic</u> <u>Context (on page 12)</u>. Consider only notes that overlap into the dissonant context by at least this set length.

#### Monophonic

Limit output to a single voice.

### Legato

Stretch successive notes to close gaps between them. Notes farther away than two bars away not considered.

#### **Tie Notes**

Merge successive notes of the same pitch into a single note.

## **Segment Voicing**

Allow individual chord segments in the **Figure** parameter to override the default voicing suggested by the **Harmony** parameter.

#### **Segment Inversions**

Allow individual chord segments in the **Figure** parameter to override the default inversion suggested by the **Harmony** parameter.

### **Harmony Alignment**

For the inversion and voicing of chords, the previous chord is taken into account if this has been set in the Harmony parameter. Deactivate this to generate all chords independently of each other.

# **Harmony Dissonance**

If set in the **Harmony** parameter, minor seconds are allowed in chords. Deactivate this to avoid dissonances.

# **Voice Leading**

Voice leading determines how melodies are supposed to be carried across multiple chord changes.

Form	Void	ce Lead	Prese	et	
Mat	erial:	Chord	s with	Extens	~
Look Al	nead:	1/12	~		
Stra	tegy:	Custor	nized		~
		V Coo	perati	ve	

# **Material**

Limits the set of notes that can be used. Rhythm guitar, for example, benefits from playing only chord notes.

# Look Ahead

How much earlier all **Figure** segments tagged with **Look Ahead** are supposed to respond to the next **Harmonic Context**. With a positive value, the next chord and scale are picked up earlier, with a negative one they are delayed.

The choice largely depends on rhythm and the sound envelope of an instrument. A phrase that pushes ahead benefits from an early response. Different values for multiple instruments can create dissonances and blurring effects.

# Strategy

- 1. None: Disables voice leading.
- 2. Default: A balanced all-purpose setting.
- 3. Adaptive: Adapts to the figure automatically by forcing rhythmically more significant notes to be harmonically stronger.
- 4. Adaptive (dynamic): Notes with higher velocity are forced to be harmonically stronger.

- 5. Adaptive (dynamic and rhythmic): Notes with higher velocity or longer duration are forced to be harmonically stronger.
- 6. **Stabilizing**: Starts with a full scale and continually increases emphasis on the chord as the measure proceeds. Obfuscates chord changes to some extent.
- 7. Solo: Affords extra leeway for melodic nuance.
- 8. Strong: Strong emphasis on chords with only few scale tones in between.
- 9. Medium: Allows more scale tones between chords.
- 10. Weak: Full scale allowed between chords. Can lead to interesting results, but may sound indifferent or dissonant at times.

### Cooperative

Follow a stricter policy to avoid conflicts with other instruments. Disable this for more dissonance and variety.

# Preset



Synfire offers an array of templates for different instruments and uses that you can recall here.

Preset

Select a predefined preset from the drop-down menu.

# **Auto-Detect**

Let Synfire examine the Figure and suggest a preset.

### Save

Save current settings as a new preset. It will appear in the menu Parameter > Insert Template.

### **Bypass**

Interpretation and voice leading are completely disabled. Symbols are rendered literally, consequences be damned.

### **Generate Pedal**

Estimate sustain pedal action from Figure (on page 294) and Scheme (on page 308). Very good for melodic piano phrases and brass solos.

### **Auto-Chords**

Automatically generate a sustained chord for every harmonic context in Harmony.

#### **Auto-Bass**

Automatically generate a sustained bass note for every harmonic context in Harmony.

# Note:

The **Express** and **Pro** editions allow for much more detailed control of voice leading, how an instrument is supposed to respond to harmony changes, the amount of dissonance that is allowed and how to deal with an instrument's playing ranges.

## **Related information**

Fine Tuning Interpretation (on page 258)

# **Morphing Setttings**

For any <u>Morphing (on page 305)</u> action to take place, two **Phrases** for the same **Instrument** need to be present in adjacent containers. Morphing thus affects the transition from one container to the next.



## Fade

You can choose between

- 1. Fade-in In the next container, the old phrase is gradually blended into the new.
- 2. Fade-out In the previous container, the old phrase is gradually fading into the new.

# **Steps**

Number of gradually different fragments to create during the cross-fade.

# Each

The length of each such fragment.

# Linear

Smoothly blend between both phrases.

# Random

Take a fragment from either phrase with gradually increasing random probability.

# Figure, Lengths, Velocities, Controllers

Include only selected parameters with the cross-fade.

# **Include Originals**

Whether to add a fragment of either original phrase at the beginning and end of the cross-fade. This increases the total length of the cross-fade.

# **Add Variation**

Adds random segment transformations to the output.

### **New Suggestion Please**

If you selected Random for the cross-fade, this throws the dice for different results.

# i Tip:

Morphing transforms the Figure (on page 294) before it is rendered. You can hear, but not see this change. If you want to keep the morphed figure, put an empty container next to it and do Container > Make Snapshot.

# **Time Inspector**

<u>~</u>	₫Ĩ		Ę	U						
Time	Time									
60	$\langle X \otimes$									
Global S	MPTE Of	fset:	F	rame Rate						
00:00	:00.00	2	4 FPS	~						
Cue Mar	kers									
00:00	04.19 E	kplosio	n	~						
Cue Tim	e:									
00:00	:04.19									
Label:										
Explos	sion									
Ма	atch with	Ruler								
📄 Rela	itive Tim	е								

Edit the details of the selected **Cue Marker** (Cue) in the <u>Time (on page 314)</u> parameter. Insert new cues with a drawing tool and position them more precisely by entering absolute time using this inspector.

### **Global SMPTE Offset**

The time code on the external equipment that matches the beginning of your arrangement.

### **Frame Rate**

FPS rate of the external video synchronized with Synfire. The frame rate has no influence on absolute time. It merely says how many frames can be addressed per second.

### **Cue Markers**

Pick an existing cue from the menu to select it and jump there.

# **Cue Time**

Edit the time of the selected cue. Format is **HH:MM:SS.FF**. The dot before the frame number is required. The time actually shown depends on the frame rate and the **Relative Time** switch below.

#### Label

Any brief description of the event or action happening at the time of the cue.

## Match With Ruler

Create <u>Tempo (on page 313)</u> changes such that the measure currently selected on the time ruler meets with the time of the cue. This makes a beat in the music play at the time of the cue.

### **Display Relative Time**

Whether to display time arrangement time, instead of absolute time in the synchronized film.

### **Related information**

Fitting Music Between Cue Markers (on page 259)

# Help Browser

# Your interactive user manual

The embedded Help Browser navigates you to places in the user manual interactively.



- 1. Open the browser with Help > Embedded Help.
- 2. Hover with the mouse over elements of the user interface to get them explained.

- 3. Select elements of the user interface to get information on the selected object.
- 4. Lock the browser with F1 (or the padlock icon) if you want to read on uninterrupted while doing something with Synfire.

Alternatively or in addition to the browser, you can enable *Help > Tool Tips* to show a small pop-up window next to the mouse for a more brief information on the item being hovered over.

# i Tip:

The help system works like a small web browser and can slow down user interface response times quite noticeably. You should close it as soon as you no longer need it to get back to normal speed.

# **Phrase Editor**

# Editing parameters in full detail

To open this editor, double-click on an instrument name, or into the empty area of a **Parameter View** (on page 144), or on a **Parameter Outlet** (on page 45).

Double-click anywhere in the empty area to close the editor.



All commands are applied to the entire parameter data if no selection is currently established. For example, you can click into a <u>Parameter View (on page 144)</u> and then use the arrow keys to transpose all values. Be sure to deselect any **Span** selection first before you do so.

# Toolbar

# £

**Span**: Selects a horizontal range of the parameter. A span with zero length shows as a vertical line and includes exactly one position.

# []

Chord: Selects one or more chords in a progression (on page 14).

# $\mathbf{k}$

**Pointer**: Selects a segment in a Figure (on page 294), or a data point in any other parameter. You can move, transpose, stretch, and compress a selection.

Press ESC to return to the last used pointer-like tool from any other tool.

# ഷ്

Symbol: Select individual symbols of a Figure (on page 294) segment.

Press ESC to return to the last used pointer-like tool from any other tool.

# 0\_

**Line**: Draws values along a straight line based on the current grid. If drawing of duplicates is disabled, grid positions may be skipped depending on the angle of the line.

Click once to add a single value.

Hold **1** Shift when you begin drawing in order to extend the nearest segment, or add parallel values to polyphonic parameters like <u>Velocity (on page 317)</u>.

Find workflow tips on drawing phrases here (on page 248).

# N

**Freehand**: Draw any curve or shape you like. Otherwise the same considerations apply as for the **Line** tool.

# %

Cutter: Split objects under the mouse.

# 

**Eraser**: Delete objects under the mouse. You can also drag open a span to delete multiple objects at once.

# 0

**Mute**: Mute individual <u>Figure (on page 294)</u> symbols you click on. You can also drag open a span to mute multiple objects at once.

# $\sim$

**Gestalt**: Shape parameter data along a curve while preserving its fine-grained structure. You can use this to gradually transpose controller data or alter dynamics over time. It works for <u>Figure (on page 294)</u> as well.

...

**Duplicates**: Enable this to allow multiple equal values to be drawn in a row. Disable it to prevent redundant controller data or repeated symbols from appearing on a horizontal line.

Double-click to remove duplicate values from the current selection or parameter.

6#

Accidentals: Enable this to allow flat or sharp symbols to be drawn with the Line tool or Freehand tool. Figure segments with accidentals will render chromatic melodic runs where the segment properties and Interpretation (on page 258) settings allow this.

Double-click to remove accidentals from symbols by rounding them to the nearest diatonic step.

#### **Grid (Outlet)**

Drop or paste any parameter here to make all subsequent drawing snap to the steps it implies. For example, drop a Figure (on page 294) to make all drawing snap to it.

Right-click to select from tuplets, combined grids and user-defined templates.

Delete it to remove the grid.

### 

Open a pop-up menu for options how to apply the grid to the current selection or parameter. Also allows the grid to be shown or hidden.

### 1, 2, 3, 4, 6, 8, 12, 16, 32

Grid: Set the grid to a fixed note length.

Double-click to quantize selected objects to that grid.

# $\infty$

Work without any grid in full MIDI resolution.

# $\rightarrow \left( \leftarrow \right)$

Snap to positions of already existing values. Useful if you want to add parallel values to a polyphonic parameter or replace them while preserving rhythm.

### H, V, C, I, P, B, R

Symbol Type: Select the type of symbol (on page 295) to draw.

Double-click to change all selected symbols to that type. If you press and hold the button, a menu opens where you can select, convert or focus symbols of this type.

# $\Xi \equiv \Xi \not\equiv$

Playing Range: Select the playing range (on page 35) for new segments.

Double-click to apply the playing range to selected segments.

# Η

**Hyper Edit**: Switch to a mode where you can edit the inherent <u>Velocity (on page 317)</u>, Length and <u>Step</u> (on page 310) of a Figure (on page 294) as if they were actual parameters.

# $\mathbf{X}$

**Zoom**: Zoom to fit all contents. Right-click or hold **Ctrl** while you click to switch between zoom memories A and B.

# Ե

**Click Board**: Open an <u>additional toolbar (on page 91)</u> for various high-level editing functions to apply to the current selection or entire parameter data.

•••••

**Text Input**: Read and write data values with the keyboard. Some parameters support this. Polyphonic values are enclosed in parentheses.

Edit existing values by selecting them with the **Span** tool. Type one or more values into the input field and the span will be filled accordingly.

# Length Input

Set a desired length of the parameter as a duration expression (on page 329).

# **Navigation**



#### **Container Caption**

When the **<u>Structure View** (on page 94)</u> is closed, a caption appears in its place, bearing the name of the selected container.

Double-click on the caption to open the Structure View (on page 94) again.

Right-click or Click-and-hold on the caption to select another container from a menu.

#### **Instrument Headers**

When the <u>Track Sheet (on page 98)</u> is closed, the list of instrument headers appears left of the phrase editor where you can select any instrument you want to edit.

Double-click on the caption to open the <u>Track Sheet (on page 98)</u> again.

#### Background

The gray button at the top of all instrument headers indicates which instrument is being displayed in the background.

Right-click or Click-and-hold on the caption to select another instrument to show up in the background.

Double-click on the caption to switch to that instrument.

# **Related information**

Editing Parameters (on page 243) Editing Figure (on page 247) Editing Harmony (Progressions) (on page 250)

# **Click Board**

Open the **Click Board** with both the toolbar. A variety of tools show up that allow for intuitive and creative manipulation of parameter data.

Most tools are shortcuts to commands that are also available on a menu or as a keyboard shortcut. Some buttons feature multiple options, popping up a menu when you keep holding the mouse button down.



This feature is available with the Express and Pro editions.

# Arrange App

# Edit one or more arrangements that belong to your project

The main Synfire window shows an <u>Arrangement (on page 19)</u>. Since <u>Music Prototyping (on page 1)</u> benefits from reusing objects, you can have multiple arrangements open at the same time and exchange data with drag & drop (or copy & paste).

# Synfire User Manual | 2 - User Interface | 92

File Edit Container Instrument Phr-	ase Parameter T	ransform	ı Library Vi	ew Playback	Window	Help									
< > 🗊 🕮 🏭			И		$\triangleright \mathscr{A}$	<b>@</b> 0	01:	00:17.04	_ 112.0 bpm	\$ © @				## III (II)	O 🗘 O 💥
[Examples/Libraries] Example Piano 🛛 🗸			Structure	Overview M	Matrix Pro	gression Ha	irmonizer Pa	lette Snip	pets Library	Sounds					
Search 🛛 🛛 🛇															9    41
▼ 🗀 Library								Ex 7 Pr	oaress °						
▼ C Factory Generated															
Calm & Aunospheric ► Chord-Less Runs		A		B Majo			А		Db Major	Drop D	Eb Minor		Gb Major		Fin °
Chords w/Melodic Chords w/Rhythm		Flute		В		Flutes	Pauses	Flutes 1		s Perc	Flutes		В	Perc	•
Counterpoint     Current in Ternary Time		Bas	son, Brass Off	Bas	ses	N	л1 (		Basses		De	scender	N	11 Descei	nder
			Ohoe	Englis	b Dro							111111	TTTTT	E	
년 Clear (3)			0.000	Ligito					D			Das	ses		
년 Entangled Lines (4) 에 Loia Roia (7)					M1							C Woodwinds	Cellos		
Meander (7)															
면 Pop E (7)															••0
Pop E (4)	I [] 🌬 🖉	<u>2</u> U K	$\otimes \circ \sim$	•••• b# 💽 🧿	ərid	✓ 1 2 3 4 5	68101216	20 24 32 \infty	୬૬ ⊞્∨્રદ્	P, B, R,	EEEZ H	[ B	20	<u>∼</u> ≣	D ₹ O
Rolling Waters (7)			19	.   10		.  11		12		13					
🗹 Walking Runs (3 Parts, Skipped)			Dur	•			•	A	47				Scheme	Interpreta	lion
앱 Walks A (4)	AB (	െള	Dm 5 (A)	C An 1 (C) 1 (A	<b>ก เม</b> ง 10	36) 5 (A)	1 (C)	AM 1 (A)	A/ -				Tempo	@@@\$	
White Pearls A (5)		- E 2				/ ,							Harmony	Form Voic	e Leading Preset
	Global Paramet		В										Time	Term Voie	Louding Hood
Abducted (Distributed) (5)														Anchors	Link within 1/4 🗸 🗸
Chord Lines (Distributed) (6)		A											Figure	Range	Fold 🗸
Chord Lines (Even) (4)		A.											Take		Octaves Strict
Chord Lines (Pour Parts) (5)														Strings	None 🗸
Chord Lines (Soft) (5)	Prime Trumpets 4 P	layers	Legato Sto	vrzato Storza		Legato	Storzato			<b>:</b>			Intemperce	Auto-Split	1/8 🗸
년 Chord Lines (w/Bass) (6)		J9		·			÷						Morphing		Monophonic
Rhythmic     Obviolate Divident													Variation		2m
▶ □ Straight Rhythin													Laver		Tie Notes
	Prime Horns 6 Playe	ers 🗸	sto			Loyaiv				1			Transnose	Chords	Enable Voicings
	7 \\\ M S	19		Slorza	10		Storzaldault	(Sustain)		12			Chromatic		Enable Inversions
Piano 013		<u> </u>													Enable Alignment Enable Dissonance
Piano.014													Pauco	Synthesis	Auto-Bass
Piano.030 *		_											Volume		Auto-Chords
Piano.032	Prime Violins 2												Pan		
Piano.033	<b>O</b> ((( M) 3)	~ b										'b	Reverb		
													Output		
													Phythm		
and the second s	Prime Cellos									: <b>   &gt;</b>					
and the the the	9 \\\ M S	19 B	Sta	Iccelo-aglig	Staco	eto egilge		Slorzato				. č	Dunamics		
												•	Valaaitu		
													relocity		

# Window Toolbar

# $\langle \rangle$

Navigate back to the previous page or return to the page you previously returned from.

# 

Open the Embedded Library, or any other Library on the sidebar.

# 

Open the Rack Module Library on the sidebar.

# ሔ

Show or hide the Structure View to make room for other views.

# Ø

Open the Phrase Editor on the current page.

# ΞΞ

Show or hide the Track Sheet to make more room for the Phrase Editor.

# $\mathbf{X}$

Zoom to fit all content in the **Structure View**, **Snippets Grid**, **Phrase Editor** or whatever currently is the main content. Right-click or hold Ctrl while you click to toggle between zoom memories A and B.

# 

Show the **Snippets View** alongside the **Structure View** so you can drag & drop phrases between both worlds.

# 

Show or hide the keyboard widget.

# 

Show or hide the right sidebar, usually the Parameter Inspector, to make room for other views.

# $\bigcirc$

Enable or disable MIDI Through.

# 2

Disable immediate audio feedback on object selection, if you want to edit more quickly.

# ?

Show or hide the embedded help browser.

# ×

Open the Audio/MIDI Setup window to configure the Global Rack and general settings.

# Structure Page

# Video Tutorial

The colored map on top is the <u>Structure View (on page 94)</u>. It shows the hierarchy of <u>Containers (on page 21)</u> that make up your arrangement. Navigate your song by selecting any container in there. The <u>Phrases (on page 17)</u> contained in it then show up on the <u>Track Sheet (on page 98)</u> below.

Right of the track sheet is the **Parameter Block** (on page 143) and the **Parameter Inspector** (on page 54) which also allows for the configuration of the selected instrument.

# Synfire User Manual | 2 - User Interface | 94



# Structure View

# Video Tutorial

This map shows where <u>Containers (on page 21)</u> are placed on the timeline. The **Root Container** at the top bears the arrangement's name and spans the duration of the entire piece. The most common situation is to have at least one container for each part of a song.

<u>P., I., I., I., I.</u>			<u>ududududu</u>	25	uhuhuhuhul	33	41	49	փոխոհանուն	57
Electro Sprikes Snippets °										
▽ Intro	Melody	B Details, ur	nmute	∽ C		D	Bassmania	Horns	Melody*	
Start	Part B °	Horns	Melody	⊽ CA	unmute	Melo	ody	B Details,	unmute	
			Part B	CAA		Pa	rt B gap!		Part B	

Select a container to make all its Phrases (on page 17) appear on the Track Sheet (on page 98).

Double-click on a container to jump to the Overview (on page 100) page.

Double-click in the empty area to close the structure view.

Drag a container to a library to save its contents.

Drop a container from a library to restore its contents.

Drag a container to another arrangement to export all its contents and sounds (Pro).

With *Container* > *New* an empty container is placed into the current container. By selecting a span on the time ruler first, the new container will be placed at that position.

# Captions

In addition to the container name, the title area of a container is showing hints that say something about its content and status.

Superscript Circle (°)

A Harmony parameter is present in the container.

### **Bold Font**

At least one parameter is provided for the selected instrument.

#### **Green Bars**

The selected parameter is in effect for the selected instrument during the indicated time span. This **Parameter Trace** (on page 97) can be disabled in the **View** menu.

### **Striped Colors And Italic Font**

The container is an alias (on page 22).

### Shaded Black Lines (\\\\)

The container is currently inactive.

### Color

You can assign a color on the **Container** tab of the **Parameter Inspector** (on page 54).

## Commands

Some of these can be found on the Container menu or the right-click menu over a container.

#### Move / Copy

Move containers around with the mouse. Hold <u>Ctrl</u> to make copies. Place containers inside others to build a structure. You can also move containers with the Arrow Keys. Hold down **Shift** to move in finer steps.

#### Resize

Resize containers by dragging them at the right edge. You can input a specific length on the **Container** tab of the **Parameter Inspector** (on page 54).

### Rename ...

Prompts you for a new name. A convenient alternative is to use the **Container** tab of the **Parameter Inspector** (on page 54).

### Add To Library

Drop a container on a Library (on page 28) to add all phrases of the container.

#### Pack

Select one or more instruments, the phrases of which you want to move to a new container. If you select a span on the timeline, only that partial range of the phrases is copied. Otherwise the phrases are moved

into the new container. Global parameters like Scheme, Tempo and Harmony are included. If you don't want them, delete them after the packing.

### Divide

Mark a position on the time ruler and do *Edit > Divide* to split a container. Child containers are also split where necessary. Where phrases need to be divided, a physical snapshot is made that is split accordingly.

### **Close Gap**

After marking a span on the time ruler, this moves all containers that are right of the span to the left. Requires the marked span to be empty, i.e. it must be a real gap. Be aware this works *inside* the currently selected container.

#### **Make Room**

After marking a span on the time ruler, this moves all containers starting at the beginning of the span or later further to the right. Be aware this works *inside* the currently selected container.

### Inactive

Deactivate a container to exclude it from rendering entirely. Great for A/B testing. Aliases may be activated or deactivated independently of their original.

### **Keep Child Containers In a Row**

Forces all children into a line without gaps, no matter how you resize or reorder them. Intended to maintain a flat song structure with successive parts.

#### **Make Alias**

Creates a read-only Alias (on page 22) of a container that you can place elsewhere to reuse its content.

### **Make Alias Physical**

Turns a read-only Alias (on page 22) into a physical copy that you can modify.

#### **Make Snapshot**

Turns all child containers and inherited (gray) parameters into a single flat container with physical copies that you can modify. Do this only if you want to significantly alter inherited parameters, or <u>condense child</u> <u>containers (on page 258)</u> to a flat structure.

### **Make Sketch**

Creates a <u>Sketch (on page 32)</u> from the selected container that you can play in real-time on **Palettes** and in <u>Progression (on page 14)</u> editors.

### **Export**

Export a **Standard MIDI File** by dragging a container to your desktop or DAW (<u>more on this here (on page</u> 270)).

# Layout

The layout of containers in the **Structure View** is computed automatically from their position and nesting. You can't arrange containers vertically by hand. The algorithmic layout makes sure the <u>rules of inheritance (on page 22)</u> are not obscured, so it is easier for you to track parameters and understand how they are affecting each other.

# Parameter Trace

Enable *View > Parameter Trace* to see a green indicator on containers indicating the presence of data for the current parameter and instrument.

1	3	5	7		9	
Masterpie	ece					>
	A°			в		
				-		
		AA °				

The trace also demonstrates nicely how parameter **Inheritance** (on page 22) works, by temporarily overriding parameter data for the duration of a sub container. In the example above, every container has parameter data present.

# Harmony Lane

The <u>Harmony Lane (on page 97)</u> provides access to the <u>Harmony (on page 300)</u> parameter of the selected <u>Container (on page 21)</u>. You can edit chords in-place using the <u>Parameter Inspector (on page 54)</u> and the menus *Parameter* and *Transform*.



Double-click on a chord or the <u>Harmony (on page 300)</u> outlet to switch to the <u>Progression (on page 14)</u> page for more details and convenience.

Chords are grayed out when there is no physical <u>Harmony (on page 300)</u> in the container. With *Parameter > Make Snapshot Of Harmony* you can insert a physical copy. Alternatively you can write chord names right into the input field of the <u>Parameter Inspector (on page 54)</u>.

# **Overview**

On the **Overview** page, the harmony lane is reduced to a mere time ruler, because at the tiny scale it is hard to discern individual chords.

# **Related information**

Harmony Concept (on page 6) Harmony Settings (on page 69)

# Track Sheet

For each **Instrument** there is a <u>Track (on page 20)</u> with a **Header** (left) and a <u>Parameter View (on page 144)</u> (right). Scroll the sheet up and down with the mouse wheel.



# **Toolbar**

Switch between three presentation modes. Hold down **C** Shift to apply the change to all containers equally.

- 1. **Open**: Phrases with at least one parameter show a full-size **Parameter View** (on page 144) and others are minimized.
- 2. Collapsed: All tracks are collapsed to a summary of the same size.
- 3. **Minimized**: Instruments without a phrase are minimized to make room. Phrases with a phrase show a summary. This is the most compact presentation.

# $\boxtimes$

Zoom to fit all contents. Right-click or hold Ctrl while you click to switch between zoom memories A and B.

# Group Buttons (A, B, C, D, E, F)

Control the visibility of Instrument Groups.

Click to view or hide a group. Alternatively you can use the menu View > Instrument Group.

Double-click to show all instruments again.

Right-click to remove all instruments from a group.

### **Headers**

### **Global Parameters**

The parameters in this phrase apply to all instruments in the container alike. This is where <u>Harmony (on</u> page 300), <u>Tempo (on page 313)</u>, <u>Scheme (on page 308)</u> and other global parameters are stored automatically, no matter where you edit them, or on which instrument you drop them.

You can experiment with non-global parameters here, like <u>Rhythm (on page 307)</u>, <u>Step (on page 310)</u>, <u>Transpose (on page 315)</u>, <u>Shift (on page 309)</u>, to achieve dramatic effects when they apply to all instruments at the same time.

#### D

Collapse or expand the **Parameter View** (on page 144). Double-click on the arrow to minimize or maximize. In **Single-Phrase Snippets Only** mode, this setting can't be changed manually in a Snippet.

### **Instrument Name**

Click to select the instrument. Double-click to open or close the **Phrase Editor**. Names appear **bold** if at least one parameter is filled with data.

### **Instrument Number (Pause)**

Three-way switch to control the <u>Pause (on page 306)</u> parameter. <u>Pause (on page 306)</u> need not be constant, though. This switch is merely for convenience.

- 1. Red: Pauses the entire phrase for the duration of the container.
- 2. Green: Overrides any pauses in parent containers and forces it off.
- 3. Gray: Neutral. Pause state possibly changes over the course of the container.

# ///

Inactivate (disable) the entire phrase. Great for A/B testing.

### Μ

Mute audio output for this instrument.

#### S

Solo audio output for this instrument.

# ß

Open the <u>Sound Wizard (on page 193)</u> to configure a sound for the instrument. There are <u>more ways to</u> assign a sound (on page 232).

### Group Button (A, B, C, D, E, F)

Controls the visibility of the Instrument Group.
Click to toggle between showing the group or all instruments.

Right-click to remove the instrument from its group.

Use the Right-click Menu over a track header, or the menu *Instrument > Assign Group* to assign the currently selected instruments to a group.

### Commands

Click on an instrument name first, to make it receive subsequent keyboard commands. You may select multiple instruments at the same time.

#### Cut, Copy, Paste, Delete, Duplicate

These commands deal with the phrase as a whole, including all its parameters. If you want to edit a parameter, select something in the **Parameter View** (on page 144) or click on an outlet in the **Parameter Block** (on page 143) first.

#### **Move Phrase**

Drag the instrument name of a track header to another container in the <u>Structure View (on page 94)</u> to move the phrase to that container. Hold down  $\frown$  Ctrl to copy the phrase.

#### **Order of Instruments**

Move a track with the keys 1 Shift + Up or 1 Shift + Down or change their order by dragging a track with the mouse.

#### **Options**

Right-click brings up the *Instrument* menu.

#### **Instrument Groups**

The purpose of **Instrument Groups** is to temporarily create sets of instruments that you want to focus on while editing your score. You may feel tempted to organize instruments by category (Woodwinds, Strings, Brass), but that's probably only useful if the groups are small enough for two or more of them to fit on one screen.



This feature is available with the **Pro** edition.

#### **Related information**

Tracks (on page 20) Global Parameters (on page 23)

# **Overview Page**

On the **Overview** page, the output of each **Instrument** is laid out as a linear track. A perspective you may be familiar with from working with a DAW. This map helps you understand how the container structure is rolling out the final music.

Every <u>container caption (on page 103)</u> indicates where at least one <u>Parameter (on page 2)</u> is present for the instrument, meaning there is a **Phrase** for it in that container.

<u>Output (on page 305)</u> is shown by default, providing you with an overview similar to a track sheet in a DAW. Link the <u>Parameter Block (on page 143)</u> to the <u>Parameter Views (on page 144)</u>, if you want to see any parameter instead of Output. For example, you can obtain an overview to see where all the Figure parameters are.



When all tracks are collapsed, you get a concise map of all phrases that you can navigate while editing them in the **Phrase Editor** (on page 86) at the bottom of the page.

### Synfire User Manual | 2 - User Interface | 102



#### **Navigate**

Double-click on a <u>container caption (on page 103)</u> to jump to the **Structure** page to edit the phrase.

#### **Select Phrase**

Click on a <u>container caption (on page 103)</u> to select a container and instrument at the same time, putting the phrase into focus.

### Ø

Open the Phrase Editor (on page 86) to edit the selected phrase.

Switch between three presentation modes. Hold down **C** Shift to apply the change to all containers equally.

- 1. **Open**: Phrases with at least one parameter show a full-size **Parameter View** (on page 144) and others are minimized.
- 2. Collapsed: All tracks are collapsed to a summary of the same size.
- 3. **Minimized**: Instruments without a phrase are minimized to make room. Phrases with a phrase show a summary. This is the most compact presentation.

Link the **Parameter Block** (on page 143) to all **Parameter Views** (on page 144) in order to display the selected parameter instead of the default **Figure** or **Output**.

### Attention:

You can't create new phrases in overview mode, unless there already is one you want to replace. This is because Synfire would have a hard time figuring out where in the hierarchy the new container should be

placed. If you want to place a new phrase, please go to the **Structure** page and add the desired container to the structure where you want it to be.

# **Container Caption**

This caption indicates where *at least one parameter* is present for an instrument. This means there is a **Phrase** for the instrument at that position. The label reveals in which container the phrase is found.

The background area without a name represents the root container.

Part B °	B Details, unmute	Part B
----------	-------------------	--------

# Commands

- Click to select the container and instrument, which puts the phrase into focus.
- Click to select a caption and press Cut, Copy, Paste, Clear, Delete to transfer the Phrase.
- Double-click to jump to the **Structure** page and into the container in order to edit the phrase.
- Drag the caption to another track to move the **Phrase** to another **Instrument**. Hold down 📉 Alt to make a copy.
- Drag the caption to a library to save a copy.
- Drop a **Phrase** on a caption to replace the existing phrase. If there is no container caption, the phrase will go to the root container.

# Matrix Page

This page provides <u>Parameter Outlets (on page 45)</u> for all **Phrases** in the selected <u>Container (on page 21)</u>. It is a good place to check for the presence of parameter data and <u>move it (on page 242)</u> between instruments and containers.

### Synfire User Manual | 2 - User Interface | 104

File Edit C	ontainer In	strument Pl	nrase Param	neter Transf	orm Library	View Play	back Wind	ow Help									
$\langle \rangle$		# ₡ 📰		K		D B	? @ O	0 🗠 0	1:00:22.23	94.0	bpm 🕹 🕓			5	3 🗰 🛄 🕻	B 🔿 🕬	@ \
				Structure	Overview	Matrix Pr	ogression F	larmonizer	Palette Snipp	ets Libra	ry Sounds						
Bulmhul																) 🗋 м	atrix
					т	he Pale R	ider (Expi	ress) °							Instrument Lab	el	7
∽ Int	ro	🗢 Chorus 1 🛛 🗢	Verse	1 ⊽¢r	orus 2 🗢 Pa	rtA ▽Re-Ir		Part B	∽ Buil	d-up		Verse	93		୍ୟାୟା Sound	i 🛄 Rang	jes
Intro P	hrases F	Chorus Phrases	Verse Phr	ases Chon	s Phrases Bass I	k Pause Intro Pt	<i>nases</i> Alte	red Phrases	Pau ses B.a	Chorus Phras	ses	/erse Pł	nrases		Sprike	Plug-in	• ď • ď
Pauses		Ohhs		С	hhs Gu	litar		Guitar	Ahh	s	 /// Off	Guitar	Ohhs		Prefer	rred Channel A Sprike AU	utc 🗸
								Ohh	s	<i>III</i>					Controlle	rs Arti	culations
					_												
Container									9 🛞								
Time	Take	Take	Take	Take	Take	Take	Take	Take	Take								
Scheme	Figure	Figure	Figure	Figure	<ul> <li>Figure</li> </ul>	Figure	Figure	Figure	<ul> <li>Figure</li> </ul>								
Harmony	Transp	Transp.	Transp	Transp	Transp	Transp	Transp	Transp	Transp.								
Reverb	Variation	Variation	Variation	Variation	Variation	Variation	Variation	Variation	Variation								
Pan	Interpr.	<ul> <li>Interpr.</li> </ul>	<ul> <li>Interpr.</li> </ul>	lnterpr.	Interpr.	Interpr.	Interpr.	Interpr.	Interpr.								
Volume	Reverb	Reverb	Reverb	Reverb	Reverb	Reverb	Reverb	Reverb	Reverb								
	Pan	Pan	Pan	Pan	😑 Pan	😑 Pan	Pan	Pan	Pan								
	Volume	Volume	Volume	<ul> <li>Volume</li> </ul>	<ul> <li>Volume</li> </ul>	Volume	Volume	Volume	Volume								
	AAS Vibraphone	NI Noire Pure	NI Noire Felt	Lead 9 (Clav)	Double Bass 3	Dreadnought Full	Pharlight (Voice)	Ahh Choir	Synfire Drums								

The Matrix console is meant as a switchboard for copying and moving parameters between instruments and containers.

- Drag any parameter Outlet to another instrument, container or library.
- Drop any parameters on the **Outlet**.
- Double-click on a Outlet to open or close the Phrase Editor.
- Drop an item from the **Rack Module Library** here (or in between the strips) to add a new **Instrument**. Drop it on an existing matrix strip to change that instrument's sound.

# Matrix Console

The Matrix console is meant as a switchboard for copying and moving parameters between instruments and containers.

- Drag any parameter **Outlet** to another instrument, container or library.
- Drop any parameters on the Outlet.
- Double-click on a Outlet to open or close the Phrase Editor.
- Drop an item from the **Rack Module Library** here (or in between the strips) to add a new **Instrument**. Drop it on an existing matrix strip to change that instrument's sound.

On the Matrix tab on the Parameter Inspector (on page 54), you can configure for which parameters you want an outlet to show up in the matrix.

# Important:

The faders for <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u> and similar controllers don't do *audio* mixing. They merely transmit MIDI messages and set the corresponding <u>Parameter (on page 2)</u> that is used to render MIDI output. This has no effect if a plug-in or device doesn't respond to the MIDI controller as expected. The master faders and **Solo**, **Mute** and **Master** buttons work at the audio level, though.

# **Progression Page**

### Video Tutorial

Edit the <u>Harmony (on page 300)</u> parameter of the currently selected <u>Container (on page 21)</u> and adjust the <u>Scheme</u> (on page 308) to match your verse metric.



# **Related information**

Editing Harmony (Progressions) (on page 250) Progressions (on page 14) Circle Of Fifth (on page 183)

# Harmonizer Page

Video Tutorial

The Harmonizer looks into one or more <u>Take (on page 312)</u> buffers or rendered <u>Output (on page 305)</u> sequences in the current <u>Container (on page 21)</u> and suggests chords that match them best. When ready, you can save the resulting <u>Harmony (on page 300)</u> parameter to the container with <u>Apply</u>.





### **Related information**

Harmonizing a Phrase (on page 254) Circle Of Fifth (on page 183)

# Palette Page

# Video Tutorial

A <u>Palette (on page 150)</u> with all bells and whistles helps you explore chords, scales and keys and their relationships, while you edit the <u>Harmony (on page 300)</u> parameter of the currently selected <u>Container (on page 21)</u>.



#### Container

At the top of the page, the name of the current <u>Container (on page 21)</u> is indicated. Right-click on it or click-and-hold to select a different container.

#### Progression

The <u>Harmony Lane (on page 97)</u> on top of the palette shows the Harmony parameter of the currently selected <u>Container (on page 21)</u>.

Drop any chords from the palette to edit the progression.

Double-click into the progression to jump to the **Progression** page so you can edit it in more detail.

#### Outlet

Drag the current progression from this <u>Parameter Outlet (on page 45)</u> to any place you want to use it. You can drop it on the embedded library on the left sidebar, for example.

Drop a progression from anywhere else to assign it to the current container.

Select this outlet and edit its contents with keyboard commands Cut, Copy or Paste.

#### Menus

These menus are the same as found on the standalone Palette App (on page 150).

Open a different palette layout with Palette > Open ... found on the left sidebar.

For more information see: Palette App (on page 150).

#### **Related information**

Playing Palettes (on page 240) Editing Palettes (on page 108) Circle Of Fifth (on page 183) Keyboard Widget (on page 185)

# **Editing Palettes**

When you edit a <u>Palette (on page 26)</u> you actually edit the <u>Scale Set (on page 11)</u> it is based on. You can save that Scale Set to the <u>Catalog (on page 16)</u> for later reuse.

# **Adding And Removing Scales**

#### 

Click on the triangle to make this scale become the **Reference Scale**. This changes the ordering of the scales.

### Add a Horizontal Scale

To start a new palette on any scale, double-click on it in the Catalog (on page 16).

- Use the right-click menu over a scale name or the *Layout* menu to add more scales to a palette.
- Drag a desired scale from the <u>Catalog (on page 16)</u> to the palette.
- Copy a scale with **#C** Control-C and return to the palette to paste it there.

#### **Add a Vertical Scale**

Vertical scales need not be included with a <u>Scale Set (on page 11)</u> in order to be available for chords. But of course you can, for example if you like its characteristic sound and want to make it a horizontal scale that starts from a particular step.

- For example, if we are in the key of C Major and want to play the scale G.phrygian-dominant (not in C Major) on the fifth degree over the chord G, open the right-click menu on chord G and select Layout > Add Vertical Scale > G.phrygian-dominant.
- Synfire automatically calculates the equivalent horizontal scale C.harmonic-minor and adds it to the palette, because C.harmonic-minor@5 = G.phrygian-dominant.

#### **Remove a Scale**

Click on a scale label and press Delete , or use the right-click menu.

#### Root

Changing the root of a Scale Set with *Palette > Change Root* ... merely transposes everything without altering its structure and chord relationships.

#### **Shifting Root**

An interesting transformation arises when the root of a palette is **shifted around**. You can, for example, pretend that the fifth be the root. Right-click on the 5th scale step and do *Assume X is Root*. The palette will be revolved correspondingly and all scales will be rotated as well.

## **Tonal Center**

You can choose a different <u>Tonal Center (on page 10)</u> with <u>Palette > Tonal Center</u>... to create an alternative palette which puts all chords, scales and note names in a different perspective.

#### **Related information**

Palette Layout & Coloring (on page 109) Playing Palettes (on page 240)

# Palette Layout & Coloring

Customize the layout, chord filter and dynamic coloring of a **Standard Palette** or **Alternative Palette** to help you explore new paths to building chord progressions.



### Layout

Edit the chord filter and layout options with *Layout* > *Settings*. Make all possible chords appear in a layout with *Layout* > *Reset Filter*. With *Layout* > *Reset Default Layout* you can restore the default layout.

### **Chord Filter Settings**

#### Catalog

Select a level of detail: Basic Chords Only, Standard, Extended or Extended + User Defined.

#### **Relationship Levels**

Minimum grade of relationship for a chord to show up:

- 1. Scale Members Only Chords that can be built exclusively from the notes of one of the scales.
- 2. **Members of Merged Scales** Likewise, but including chords that can be built from the union of all scales. When available, these chords appear in the lower part of the palette.
- 3. **Partial Members with Function** Likewise, but including chords that have at least one note in common with a scale and have an essential harmonic function in the tonal center. When available, these appear hanging from the top of the palette.
- 4. **Non-Members with Function** Likewise, but including all remaining chords that have a harmonic function in the tonal center. When available, these appear hanging from the top of the palette..
- 5. All No restrictions. All chords is displayed.

#### **Modifications**

A list of options you have set for individual chords from the right-click menu over a chord. This list is for information and selectively deleting options.

#### **Functions (Display Detail)**

Level of detail for harmonic function labels.

#### **Added Functions**

Add arbitrary harmonic functions to the palette, typed as <u>function expressions (on page 319)</u> or as a <u>chord (on page 327)</u>.

### **Added Chords**

Add arbitrary chords that do not appear in the palette by default, like more remotely related functions. If Synfire recognizes the chord as a valid harmonic function, it will be added to the **Added Functions** list.

### Troubleshooting:

If a chord has been added but does not appear in the palette, this may be due to the current filter or the other layout settings.

#### Layout

Other layout settings

- Names, Numerals or Functions Determines the way in which chord names are displayed. In Function mode, chords that have no significant function are filtered out, reducing the number of chords significantly.
- 2. Enlarge Basic Functions T, D, S Emphasizes the primary functions Tonic (T), Sub-dominant (S), and Dominant (D) by enlarging the chord boxes.
- 3. Group Chords Similar chords are grouped into a common box. Greatly improves clarity.
- 4. Label Scales With Notes Note names are used to label the center line instead of Roman numerals.

#### **Margins**

The margins between boxes.

### **Including And Excluding Chords**

Besides using the layout settings dialog, there are more ways to include chords with a palette, or exclude (hide) them.

#### **Including Chords**

- 1. Add a chord by name or function as described above.
- 2. Drop a chord on the palette from another palette or progression.
- To show chords that have previously been hidden, select any chord at the desired scale degree and choose Layout > Add Chord or Layout > Add Chord Outside Scale Set from the right-click menu respectively.
- 4. Use *Layout > Progression On Clipboard > Include All Chords* to include all chords of the current progression on the clipboard.

#### **Excluding Chords**

Hide all chords that you find unnecessary, distracting or can't play.

- 1. Select one or more chords in the palette and hide them with *Edit > Delete*.
- 2. Hide all chords of a particular type with Layout > Remove All Chords of Type X.

#### **Marking Chords**

Highlight chords with a color band, so you can better remember them, or to draw the attention of others to a distinctive spot on your palette.

- 1. Select the desired chords in the palette and use the menu *Layout > Markings* to add or remove color markings.
- 2. To mark all chords of a the same type, hold down **1** Shift when picking the above mentioned menu item.
- Use Layout > Progression On Clipboard > Marker Chords to mark all chords of the current progression on the clipboard.

# **Coloring Settings**

The current **Coloring Scheme** of a Palette can be examined and edited with **Coloring > Settings** .... Several built-in coloring algorithms can be placed in a sequence to compute the final color of a chord from the top down.

Mapping Sequence	Algorithm Settings				Backgrour	nd
Type: Dominant Type: Major Type: Minor	✓ Active ■ Transparent				🗸 Transpa	rent
Type: Other Type: Suspended Function Grade Scale Membership Subtriads Subsets	Weak ISB H, S, B RGB	Reverse Scroll	Strong H, S, B		HSB RGB	н, s, в
Continuation	Blend Rule	Description				
Relationship Fragility History	Brightness ✓ Amount 0,83 Offset -0,83 Fixed Reference	Continuation Highlights ct to follow the according to thousands o chords are ta	nords that are mor currently playing o a statistic derived f songs. The previo aken into considera	e likely chord from pus two ation.		
	ок	Apply	Cance			

#### **Mapping Sequence**

Select an algorithm. Use the up and down buttons below the list to change the position of an algorithm in the sequence. Color is computed from the top down, each algorithm processing the color of the previous one.

#### Active

Enables the selected algorithm. A disabled algorithm is bypassed.

#### Transparent

Makes the previous color pass through unchanged where the algorithm delivers an undefined value.

#### **Color Gradient Settings**

Defines how values 0.0 ... 1.0 are mapped to a color. Select RGB or HSB color space and set the beginning and end values of the gradient. Use **Reverse** to reverse the gradient and **Scroll** to rotate it.

#### Amount, Offset

Multiplies the value of the algorithm by an **Amount** and adds an **Offset** to it, before it is mapped to the color gradient. Not all blend rules make use of these settings.

#### **Blend Rule**

Determines how the color of the previous algorithm is combined with the color of the current one.

- 1. Blend: Blend both colors in a ratio set by the Amount slider.
- 2. Brightness: Modulate the brightness of the previous color, using the Amount and Offset sliders.
- 3. Saturation: Modulate the saturation of the previous color, using the Amount and Offset sliders.
- 4. Hue: Modulate the color hue of the previous color, using the Amount and Offset sliders.
- 5. Replace: Replace the previous color with the current, unless it is undefined.

#### **Fixed Reference**

Defines a particular chord as the permanent input to the coloring scheme. That is, the coloring stays put and won't follow the chords being clicked. Select the desired chord in the palette *before* you open the dialog and enable this setting.

### *i* Tip:

Fixed Reference is a way to permanently freeze a coloring you happen to find useful.

### **Coloring Algorithms**

The *Coloring* menu can be used to enable or disable any of the built-in coloring algorithms, all of which are listed below.

#### **Static Coloring**

These coloring algorithms provide useful background shading for orientation and don't update with a new chord.

#### **Type: Suspended**

Highlights suspended chords with shades of green.

#### **Type: Dominant**

Highlights dominant chords with shades of red.

#### **Type: Major**

Highlights major chords with shades of yellow.

#### **Type: Minor**

Highlights minor chords with shades of blue.

#### **Type: Other**

Highlights other chord types with shades of cyan.

#### **Function**

Highlights chords with a strong functional significance. Primary functions Tonic, Sub-dominant, and Dominant are considered strong. Secondary functions appear dimmed. Chords lacking a clear function remain dark. This scheme provides orientation, especially in <u>Alternative Palettes (on page 27)</u> with an unfamiliar structure.

#### Grade

Indicates the grade of relationship a chord has to the <u>Scale Set (on page 11)</u> of the palette. The brighter a chord, the closer its relationship to the <u>reference scale (on page 11)</u>.

#### **Scale Membership**

Each scale of the <u>Scale Set (on page 11)</u> is associated with a color. A chord shows up in a color only if it can be built exclusively from the notes of the associated scale (and no notes from other scales). This highlights all chords that may best emphasize the characteristics of a particular scale.

#### **Dynamic Coloring**

These coloring algorithms highlight some relationship with the chord currently playing.

#### **Sub-triads**

Highlights all simple triads that are completely contained in the currently played chord. You can play these triads as simplified substitutes for the current chord, or achieve <u>polytonality (on page 15)</u> by letting multiple instruments each play a different sub-triad using the <u>Layer (on page 302)</u> parameter.

#### **Subsets**

Highlights all chords that are completely contained in the chord currently being played. This allows you to find several smaller chords that would make up the current chord if played together.

#### Supersets

Highlights all chords that contain the current chord in full. This allows you to find more complex chords that are suitable as an extension to the current chord.

#### Intersection

Highlight appears brighter the more notes a chord has in common with the currently playing chord. If you click along a path of brighter chords, the sound will only change gradually. Conversely, you can achieve a stronger contrast by following a path along darker chords.

#### Relationship

Highlights chords whose <u>harmonic function (on page 319)</u> is closer to the function of the currently playing chord.

#### Strength

Highlights chords that, when played next, introduce a strong change from the currently playing chord. A short distance in the circle of fifths is considered strong. Longer distances are considered weaker. For example, you could follow the bright path for a chorus and the dark path for a verse.

#### Continuation

Highlights chords that are more likely to follow the currently playing chord according to a statistic derived from thousands of songs. The previous two chords are taken into consideration.

#### **History**

Highlights the recently visited chords as a slowly fading trail of green color to help you remember what you were playing.

#### Tonnetz

Chords related to the current chord are highlighted according to the Tonnetz (Tone Network) theory. Neighbors S, N, L, R, P each show a distinct color. Learn more about the Tonnetz on <u>Wikipedia</u>.

### **Saving Presets For Layouts**

You can save a layout as a preset with *Layout > Preset > Save As* .... Presets you saved to the folder **Layouts** in the **Configuration Folder** will appear in the preset menu.

# **Saving Presets For Coloring Schemes**

Save a coloring scheme as a preset with *Coloring > Preset > Save As* .... Presets you saved to the folder Colorings in the Configuration Folder will appear in the preset menu.

# i Tip:

You can save the current layout and coloring scheme in combination as the system default with *Palette > Default Filter and Coloring > Save*. This command is available only on the File menu of a standalone palette editor.

# **Snippets Page**

### Video Tutorial

The **Snippets View** is home to **Snippet Groups**, each of which provides multiple slots for <u>Snippets (on page 24)</u>. You can populate these slots simply by dropping **Phrases**.

### Synfire User Manual | 2 - User Interface | 116

File Edit Container Instrument Phra	ase Parameter Transf	orm Library View Play	/back Window Help			
< > 🗊 🖾 🖧 📰	K		2 🙃 🔿 🗠 01:	00:06.11 94.0 bpm	¢ C Q E	ೱ Ⅲ Ⅲ □ 😷 ᠲ) ⊘ ※
[Examples/Libraries] Factories List 🛛 🗸 🧭	Structure	Overview Matrix Pr	ogression Harmonizer Pa	lette Snippets Library	Sounds	
Search 🛛 🖸 🗸	< > 🕑 🕸	S A B	C D E F	G H Harm	12 x 8 💙	😑 🗹 🔟 Group
▶ 🗋 Lines.Bass	😂 Harmony	🕞 Cmaj7 G7 Em7 🕞 Am(a	dd9) Gma 🕨 G C F D7 • 📄	666 *		Instrument Label
▶ 🗀 Lines.Chords	· · .					
► Cirles.Melody ► Parameters	AAS Vibraphone	🕞 Intro Phrases 🕨 Verse	Phrases 🕞 Altered Phrases			-III Sound III Ranges
✓ ☐ Manually Edited ⊗ <sub>B</sub> Auto-Bass	·᠁ NI Noire Pure	D Intro Phrases D Choru	s Phrases 🕨 Verse Phrases			Audio Plug-in
® Auto-Chords	📾 NI Noire Felt	Intro Phrases Verse	Phrases			Sprike Y 🗹 01 Lead 9 (Clav) Y 😚
√ Shift: Soft Humanizer	'빠이 Lead 9 (Clav)	Chorus Phrase				Preferred Channel Autc 🗸
	Bouble Bass 3	Chorus Phrases > Verse	Phrases Dass & Pause			Sprike AU 🖉
	· · · · · · · · · · · · · · · · · · ·					Controllers Articulations
✓ Flow: Metric Stops ✓ Flow: Rhythmic Bass	Creadnought Full	Ditar Ditar				
Parameters.Harmony     Parameters.Pause	'빼치 Pharlight (Voice)	Ohhs				
Parameters.Skip     Parameters.Step	📾 Ahh Choir	Ahhs				
Parameters.Velocity     Dhracos Counterpoint	Synfire Drums	🕞 Intro Phrases 🕨 Choru	s Phrases ▷ Verse Phrases ▷	Off Diff Diff	es 🕞 Build-up 🕞 B.a	
► Phrases.Counterpoint ▼ □ Phrases.Drums						· • •
🖻 ExperimentalKit1			a _ a _ a			
	Live 1		4 5	6 7		
	O Volume	🔵 Volume 😑 Volume	Volume OVolume	😑 Volume 😑 Volume	O Volume	
	😑 Pan	🦰 Pan 💿 Pan	Pan Pan	Pan Pan	Pan Pan	
	Reverb	Reverb Reverb	Reverb Reverb	Reverb Reverb	Reverb Reverb	
	Pause	Pause Pause	Pause Pause	Pause Pause	Pause Pause	
	S N	S M S	M S M S M	S M S M	S M S M	
Loop Piano.001	5					
		í 🛓 í 📋	ĭ ⊥ĭ í ĭ			
· · · · · · · · · · · · · · · · · · ·		NI Noiro	Load 9 Double Perce	Dreadpought Dhavilet	Sunfire	
	Vibraphone	Pure NI Noire Fe	t (Clav) 3	Full (Voice)	Ahh Choir Drums	
╸╴ <sup>╸</sup> ╺╼╼╼╴╴╴						

Learn more about Snippets (on page 24).

### Toolbar

# $\langle \rangle$

Navigate back to the set of snippets you had playing before.

# Ø

Schedules all currently playing snippets to be stopped next time you press the launch button. Use this when you want to schedule an entirely new set of snippets from scratch.

# ۵.

Schedule a random selection of snippets to play next. Hold down **1** Shift to keep the current Harmony unchanged.

# 8

Launches all armed snippets. If you want to start or stop multiple snippets in synch, arm them with a right-click on their play icon first and then click this button to execute the changes at the same time.

# A, B, C, D, E, F, G, H

Memories holding the current state of playback. Right-click to save, left-click to recall. Hold down Shift to exclude Harmony from the recall.

Right-click while holding H Control to clear a memory slot.

Drag a memory button to a container on the Structure page to insert it into your arrangement.

Drag a memory button to a library to save its phrases and current playback state.

Drop a memory button from a library to restore its previous playback state. This is only useful for short term backups though, as snippets that have been deleted will not be restored automatically. You restore snippets manually by dropping the individual phrases on the snippet grid.

Rearrange memory buttons with the mouse. Their values are swapped.

#### Harmony (Outlet)

Drop a <u>Harmony (on page 300)</u> parameter here to use as the default chord progression when no currently playing snippet provides one.

#### **Grid Size**

Shrink or enlarge the grid.

=

Change orientation of the grid.

```
\square
```

Options Menu:

- Single-Phrase Snippets Only: Enable this mode if you want all groups to play a single instrument only. Groups won't accept containers anymore and Synfire will less frequently ask you questions what to do when you dropped something on a slot.
- Make Snippets From Arrangement: Collect phrases from all containers and create a snippet for each.
- Make Arrangement From Last Session: Assemble a new arrangement from the playback you just stopped.
- Mark/Unmark Used Snippets: Earmarks all snippets that are currently used by any memory slot. Helps you make room for new content without accidentally deleting important snippets.
- **Profiles**: All current probabilities can be saved in a profile or appended to an existing profile. Probabilities are associated according to the respective category of an instrument.
- Clear Snippets: Delete all snippets.

#### Transport

The transport buttons on this page operate live snippets playback exclusively. If you want to start or stop the current arrangement, you need to switch to the **Structure** page.

 $\triangleright$ 

The play button starts or continues live snippet playback. You won't hear anything until at least one snippet is scheduled for playback.

The stop button pauses current live playback. Clicking the stop button again clears all currently scheduled snippets so you can start over from scratch.

Save audio output of your next session to a file. The file is placed next to the arrangement document (*File > Show File in Finder*).

### Console

The console on the **Snippets** page controls the live mix to some extent, provided your plug-ins respond to the respective MIDI controllers.

#### **Export Snippets As Audio Loops**

You can export individual snippets, snippet groups or the entire grid as audio loops to disk ("Clips") to import them into your DAW. You find this option on the right-click menus of snippets and groups, and the options menu of the grid.

Select a Snippet with Harmony first. Its progression will be used to render all snippets in order to make them compatible with each other (unless a snippet contains its own Harmony). The audio files are placed next to the arrangement document (*File > Show File in Finder*).

You can cancel a running export at any time with Stop. You should avoid manual interventions while an export is being processed, as all audio output is recorded as it goes into the clips. The audio files are trimmed only after the last snippet has been captured. Do not use any files before this process has completed.

Note:

Exported snippets are trimmed to a multiple of the length of Harmony. They may loop more than once if they are shorter than this. This ensures all clips are harmonically compatible with each other when you play them in your DAW's grid live.

Note:

This feature is available with the Express and Pro editions.

#### **Related information**

Snippets Console (on page 122)

# **Snippet Group**

A **Snippet Group** bundles a number of **Snippets**, each of which contains parameters or **Phrases** for one or more instruments. Only one snippet in a group can play at a time. A group shows up in the first column of the grid as a button with the group's name.



Drop a **<u>Phrase Pool** (on page 28)</u> on a group to create a snippet for each favorite phrase in the pool.

Drag a snippet group up or down to re-arrange groups on the grid.

Right-click or click and hold on a group to open the options menu.

### **Options**

#### Rename

Rename the group.

#### **Set Color**

Set a color for the group.

**Make Snippets From Harmony** 

Collect from the current arrangement all <u>Harmony (on page 300)</u> parameters and add a snippet for each to this group.

#### **Make Snippets From Instrument**

Collect all phrases for an instrument from the current arrangement and add a snippet for each.

# **Types**

A group automatically changes its type depending on what is currently included with its slots.

The group is empty and ready to take phrases and parameters that you can drop on any of its slots. Unless **Single-Phrase Snippets** are <u>enforced (on page</u>), the group will also take containers with phrases for multiple instruments.

#### 夏

The group includes snippets with phrases for a single instrument only. If you want to assign an instrument to a group, simply drop a phrase on an empty slot to create a first snippet. The group will assume the instrument associated with the phrase.

### $\mathfrak{S}$

Indicates a **Mixed Group** that includes snippets with phrases for multiple instruments, or global parameters that affect all instruments.

# **Group Inspector**

When a snippet group is selected, you can edit its name, color and other properties on the right sidebar.

Group	<u>~</u>	顚								
Snippet Group										
Dubstep	Dubstep Pattern									
Change			ත් ( 	∽ %						
Silence			40	%						
Global I	Modula	tion								
Change			1,0							
Silence			1,0							

#### **Probability Of Change**

When rolling the dice, this is how likely the playback state in this group will change. If you want to control a group manually, you can disable this by selecting zero percent.

#### **Probability Of Silence**

When rolling the dice, this is how likely all snippets in this group will go silent. Set to zero percent if you want this group to play one of its snippets all the time.

#### **Global Modulation**

Multiplies each group's probabilities with a factor you can set here. If you increase the silence factor, your random arrangement will, on average, get more sparse and less dense. If you increase the change factor, more snippets will be replaced on each turn, leading to more drama and variety. This setting is global. It shows same values no matter which group is selected.

# Snippet

The slots of a snippet group are initially empty. Once you drop something on them, a **Snippet** will be created and show up in the slot.

# Electronic

Drag snippets to re-arrange their order in a group. Moving a snippet to another group will ask you whether to also move the phrase to that instrument, or make a **Mixed Group** with multiple instruments. Hold down Ctrl while you drag to make a copy.

Drop a snippet, a regular container, a phrase, a parameter or a (partial) selection on a slot to create a <u>Snippet (on page</u> <u>24)</u> or replace its current content.

Double-click on an empty slot to create an empty snippet.

Double-click on a populated snippet to edit its phrases on the Structure (on page 93) page.

Cut or Copy a snippet to insert it elsewhere.

Paste a snippet, a regular container, a phrase, a parameter or a (partial) selection to populate the slot. For example, you can copy a range of chords from a progression and paste it here, or drag and drop that selection with the mouse.

Click on an empty snippet cell to select the root container of all snippets. It contains three parameters **Tempo**, **Scheme** and **Harmony** only. The harmony parameter is used as a default in case no currently playing snippet provides one.

Drag a snippet to a library to backup its contents.

Drop a snippet from a library to restore its contents.

# $\triangleright$

The triangle is used to control playback of the snippet.

Click to schedule the snippet for playback when the next measure is due.

Right-click to arm the snippet to be scheduled for playback the next time you click the launch button so the toolbar. You can use this to arm any number of snippets in advance and then schedule all of them at the same time.

# **Snippets Are Containers**

Since a Snippet is basically the same as a **Container**, you can include with it any number of **Parameters** (on page 2) for any number of **Instruments**. This way you can place all sorts of sections and parts on your snippet grid and experiment with them live.

In **Singe-Phrase Snippets Only** <u>mode (on page</u>) however, all Snippets can take parameters or phrases for a single instrument only. While this is easier to handle, it doesn't allow you to arrange parts and sections in real-time.

# **Snippets Console**



The small console on the **Snippets** page controls the live mix to some extent, provided your plug-ins respond to the respective MIDI controllers.

• Drop an item from the **Rack Module Library** here (or in between the strips) to add a new **Instrument**. Drop it on an existing matrix strip to change that instrument's sound.

### Important:

Arrangement containers and snippets share the same instruments. That is, these are identical. If you add or remove instruments on either page this will be mirrored on both.

#### Important:

The faders for <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u> and similar controllers don't do *audio* mixing. They merely transmit MIDI messages and set the corresponding <u>Parameter (on page 2)</u> that is used to render MIDI output. This has no effect if a plug-in or device doesn't respond to the MIDI controller as expected. The master faders and **Solo**, **Mute** and **Master** buttons work at the audio level, though.

# Library Page

#### Video Tutorial

Here you edit the **Embedded Library** (on page 23) owned by this arrangement. You also have the option to open and edit any other Library (on page 28), if you don't want to use the **Standalone Library Editor** (on page 149).



Note: This feature is available with the **Express** and **Pro** editions.

# **Phrase Pool Editor**

The **Phrases** contained in the selected **Phrase Pool** (on page 28) are listed on top of the page. Click on any phrase to select it and load it into the editor.

### Synfire User Manual | 2 - User Interface | 124

		Editor	Harmonizer Figure	Recognition Factory	·			
Loop	Piano.001	Piano.002 🗢	Piano.003 *	Piano.004	Piano.005	Piano.006	Piano.007	
	Piano.008	Piano.009	Piano.010					
ST Clabal								
Global	v 🕒 Scheme	<ul> <li>Figure</li> <li>Figure</li> <li>Velocities</li> </ul>	crim	Rhythm S	kip Output	Interpretation	Bend Expression	n 🕒 Sustain
Preview	Tempo	Take Len	gth Step	Shift			Breath Modulation	n Portamento
£ []	< < 2 < < < < < < < < < < < < < < < <	∑ [?] へ   💽 Grid	1 2 4 6 8 12	: 16 24 32 ∞ → ← 🖂	НУСРВ	R. = = = = =	⊮ HX50	<b>2/1</b>
$\downarrow \uparrow$			*3 ↓8 ↑8 (		<b>№ 5 %</b> 1 %	ia. 6° ## J-	$J + [A] \rightarrow ]$	50
Cr	nai7	G7	Am7	Em7	Fmai	7	Cmai7	Fmai7
1								4
				₽ ∕_₽	-			
6								
4			=>					
3								
-1								
-2								
-3 -4								
-5 -6								

The pages **Editor**, **Harmonizer** and **Figure Recognition** exist for your convenience, each providing a focus on a specific task that is involved with editing a pool. Under **Factory** you can generate new phrases.

### **Preview (Outlet)**

Drop any <u>Harmony (on page 300)</u> here to make it the **Preview Progression** of the pool. It will be used to play all phrases, unless a phrase comes with its own <u>Harmony (on page 300)</u> parameter.

Pick a template from the right-click menu.

# 60

Preview the selected phrase using the **Preview Progression** of its pool, or its own <u>Harmony (on page</u> <u>300)</u> parameter, if one is present.

# Sidebar



# <u>~</u>

Show the **<u>Parameter Inspector** (on page 54) for the selected parameter. Alternatively, double-click on a parameter outlet.</u>

# **(i)**

Edit meta data and properties of the selected phrase.

#### Name

Any name you find helpful.

### Density

Estimated value for sorting phrases by the number of notes in them per measure.

### **Master Take**

Marks the phrase as a resource that was obtained from import or recording. Such a phrase is usually very long and the actual phrases of the pool get extracted out of it by splitting it into pieces.

#### **Default Phrase (Preview)**

Play this phrase when a pool is selected as a whole in the library browser.

#### **Favorite**

Mark the phrase as a favorite. You can later delete all other phrases in one go with *Pool > Keep Only Favorites*.

#### Usage

This is a suggestion what role the phrase might typically take.

- 1. **Opening**: Good as an intro for building up tension.
- 2. Loop: Good for being repeated over and over, like a riff.
- 3. Bridge: Good for bridging two parts, introducing a change.
- 4. Closing: Good for bringing a part to a close.
- 5. Temporary: To be honest, this is a pretty useless item that I should be beaten to.

#### **Related information**

Phrase Pools (on page 28)

# **Editor Tab**

Switch to this page to edit an existing phrase or cleanup an imported phrase for subsequent **Harmonization** and **Figure Recognition** on the next two pages. Select the phrase you want to edit from the grid above.

Show the Parameter Inspector (on page 54) or double-click on any parameter Outlet.

# **(i)**

Edit a phrase's properties.

### **Typical Tasks**

- Remove parameters that are not needed for a reusable Phrase.
- Trim and tag Take (on page 312), in case you still need to run Figure Recognition.
- Trim and optimize Figure (on page 294) to make it as reusable as possible.
- Assign a Preview (on page 307) progression to the pool.
- Assign a Sound to the pool.

### **Related information**

Phrase Editor (on page 86)

# Harmonizer Tab

Switch to this page after you have cleaned up (and possibly tagged) the phrase's <u>Take (on page 312)</u> for harmonization. Here you estimate a <u>Harmony (on page 300)</u> parameter based on the <u>Take (on page 312)</u> of the currently selected **Phrase**.

The presence of a <u>Harmony (on page 300)</u> parameter is a prerequisite for **Figure Recognition**, which you can perform on the <u>next page (on page 127)</u>.

i Tip:

Short fragments do not lend themselves to good harmonization, since they contain too few notes. You will get better results from long takes, preferably an entire track worth of imported MIDI data.

#### **Related information**

Harmonizing a Phrase (on page 254)

# **Figure Recognition Tab**

Switch to this page after you obtained a <u>Harmony (on page 300)</u> parameter from the previous page. This page is basically the **Phrase Editor**, only hard-wired to show a window layout that is best for the task. Perform a <u>Figure</u> <u>Recognition (on page 74)</u> as always, until you are satisfied with the result.

Now your phrase is ready for use.

### **Related information**

Figure Recognition (on page 5) Take Settings (on page 74)

# **Factory Tab**

Switch to this page if you want to generate random phrases from scratch, using one of several KIM Factories, or generate new variations of existing phrases that have been generated previously.



Select a <u>Factory (on page 29)</u> from the pop-up menus in order to generate new **Phrases**, or select an already generated phrase to recall its original factory settings. Every generated phrase is basically a new Factory, of which you can generate more variations, or alter its settings and spawn off a new series of different phrases.

#### **Bundles**

Select from this pop-up menu a KIM Bundle of factories.

### **Factories**

Select from this pop-up menu a KIM Factory from the current bundle.

#### **Factory User Interface**

Each Factory has a distinct user interface that reflects its structure. Think of it as a pre-wired modular synthesizer. Navigate the structure of the **Factory** using the tabs.

# i Tip:

Open the Help Browser for information on each Factory.

#### ٣

Generates a new phrase based on the current settings. The generated phrase is added to the current pool in the library. When you switch to the **Structure** page (or any other page), you will find it on the **Phrase Library Sidebar** (on page 49), where you can drag it to anywhere you want to use it.

#### $\heartsuit$

Marks a phrase as a favorite. With *Pool* > *Keep Only Favorites* you can eventually cleanup a pool to only retain the phrases you marked this way.

### Û

Deletes the currently selected phrase.

#### Important:

Every generated phrase retains the settings used to generate it, so you can return to it later and continue to spawn off new variations. A carefully configured generated phrase therefore is a new **Factory** in its own right.

# i Tip:

You can edit a generated phrase in any way you choose. For example, you can transpose melodies or delete symbols to make more room for other instruments.

# Note:

This feature is available with the Pro edition.

#### **Related information**

Factories (on page 29)

# Sounds Page

The **Arrangement Rack** is hosting **Sounds** for the **Instruments** of the arrangement. **Device Descriptions** are copies included with the file in order to make it self-contained and independent. You can edit these devices here.



Enable View > Library Sounds to see the racks of all currently open Libraries.

# **Rack Editor**

There are three types of racks. <u>Arrangement Racks (on page 24)</u>, <u>Library Racks (on page 149)</u> or the <u>Global Rack (on page 39)</u>. Their <u>Rack Modules (on page 38)</u> (aka Modules) appear listed in this view.

Arrangement 🔚 Arrangement Rack 🖉 Device Descriptions 🕢 Repository	
All MIDI Embedded Engine1 Ø № 🖽 C 🗄 🖸	
© Engine1:04 ✓ Kontakt 5 AU General Purpose:A Bant ✓ ▷ Insert ▷ Insert ○ Gain O B O S M General Purpose:A Band	Rack Module
Engine1:05     Kontakt AU       Kontakt Pharlight     Insert         Gain     AUX Send       S     M       Pharlight     Insert	Device Description: General Purpose:A Band * 2 variants, 27 sounds on fixed channels
Image: Soundcase FluidR3       ✓       Soundcase AU       Sain       AUX Send       S       M         Soundcase FluidR3       ✓       ▷       Insert       O       O       O       O       Soundcase FluidR3       Soundcase FluidR3	1 ♥ Classic Bass
	3       Contemp Analog - Sub f         4       E-Piano         5       EP 10-Piano         6       Jazz Guitar         7       Rock Guitar         8       Jazz Organ         9       Muted Trumpet
	10 V Street Knowledge Kit
	11     V     Drum Machine-CR-78       12     V     Toys-Rapman Vibraphor       13     V     String Melody - Orchest

### **Window Toolbar**

# 

Opens the **<u>Rack Module Library** (on page 47)</u> on the left sidebar where you can search for **Rack Module Presets** and other items and drop them on the rack to add a new <u>**Rack Module** (on page 38)</u> or load a plug-in and device into an already existing module.

### **Rack Toolbar**

#### All, MIDI, Embedded, Engine1

Select one of these to focus on the modules hosted on that type of port. This selection also determines which host a new module is added to when you double-click in the empty area of the rack.

# 0

When active, the arrangement remembers the **<u>Global Rack</u>** (on page 39) that is currently open. Next time you open the arrangement, you will be asked whether to also load that global rack.

# **1**

Loads the **Global Rack** (on page 39) that was open at the time this arrangement was last saved.

Open the current <u>Global Rack (on page 39)</u> in the <u>Audio/MIDI Setup (on page 156)</u> app.

# G

Reload all rack modules. This may help to restore a more stable state if something went wrong.

### ΨĒ

Sort all rack modules by type and port.

### $\mathbf{\nabla}$

Opens a pop-up menu to control the **Audio Engine** (on page 41) and purge unused modules, devices or ports.

# Selection, Drag & Drop

- Select a module to see its <u>contents on the right sidebar (on page 133)</u>. Select a channel on the sidebar to open a **Keyboard Widget** where you can play a sound and examine its **Playing Ranges**.
- Drag a module to another rack to copy it.
- Drop an item from the <u>Rack Module Library (on page 47)</u> to insert a new module or load a plug-in and device into an already existing module. You can drop items between the modules.
- Drop an effects plug-in or effects preset from the <u>Rack Module Library (on page 47)</u> on an existing module to load it as the insert effect.
- Move modules up or down to change their order.

### **Related information**

Rack Module (on page 131) Sidebar: Rack Module Library (on page 47) Device Inspector (on page 133)

# **Rack Module**

Setup a Rack Module (on page 38) to provide the Sounds (on page 37) you need.



#### **Audio Port**

Moves the module to another host or port. Select an **Audio Engine Port** or **Drone** as the target where you want the module to be moved. Learn how to move a module to a Drone in a DAW <u>here (on page 239)</u>.

#### **Output Port**

Pick an external MIDI port for the module to send its output to (MIDI modules only). Any port may receive output only from a single module.

#### **Device Description**

Select a known **Device Description** that matches the plug-in or external hardware, or perform one of these actions:

#### Detect

Matches the current state of the plug-in with previously saved **Rack Module Presets** and select the device description associated with those.

#### **Unknown Device**

Go without a device description for now.

#### **Create New**

Create a new device description (on page 234) for use with this module.

#### Extract From Plug-in ...

Attempt to extract a device description (on page 236) from the plug-in.

#### Download From Repository ...

Search the **Online Repository** (on page 182) for a device description that matches the current plug-in.

#### **Show Device Layout**

Opens an inspector showing the channel layout of the device you selected for the module.

### **Show Insert Module**

Opens an inspector for you to setup an **Effects Plug-in**. One insert effect per module can be chained *after* the plug-in's output. A device description is not needed unless you want Synfire to be able to select different programs.

# i) Tip:

Racks offer basic audio processing only. For production and mixing you will want to move your project to a DAW.

#### **3-Way Plug-in Button**

This button allows for three tasks to be performed:

- Load or save a <u>Rack Module Preset (on page 38)</u>. New presets are listed in the <u>Rack Module</u> <u>Library (on page 47)</u> automatically.
- 2. Open the current **Plug-in Editor**.
- 3. Load an Audio Plug-in.

#### Gain

Adjust the overall audio level for this module. Don't use this for mixing. Keep at 0 dB unless you need to compensate for grossly unequal audio levels.

#### **AUX Send**

Send a portion of the output to the AUX Module (on page 39), if the Global Rack has one.

#### Solo

Temporarily mutes all other rack modules.

#### Mute

Temporarily mutes only this rack module.

#### **Colors, Icons**

You can customize a rack module to improve your oversight.

#### Label

Enter a name for your rack module.

#### **Option Menu**

#### **Exclude From Rack**

Detaches the rack module from the current rack, so you can attach it to another. Makes it a <u>Transient</u> <u>Rack Module (on page 39)</u> that is no longer saved with the current rack.

#### **Include With Rack**

Attaches a Transient Rack Module (on page 39) to the current rack, making it a permanent part of it.

# Commands

- Drag a rack module to another rack to copy it.
- Keyboard shortcuts for Cut, Copy, Paste or Duplicate also work on a rack module.

#### **Related information**

Special Rack Modules (on page 39) Rack Module Preset (on page 38) Sidebar: Rack Module Library (on page 47) Relocating From Engine to DAW (on page 239)

# **Device Inspector**

Shows a brief overview of the **Device** used in a **Rack Module**, listing its channels and which sounds are selected and used at this time.

Select a channel to make a keyboard controller appear, allowing you to play or preview the sound.



Ø

Jump to the device editor for more details, or in-depth editing of the Device Description.

# **Devices Page**

Under the tab **Device Descriptions** you manage your collection of **Devices** (on page 36). Devices managed inside an **<u>Arrangement</u>** (on page 19)</del> are a part of that arrangement. They can't be used by other arrangements unless you copy them with **Device > Save to Global Rack**.

Devices managed by Audio/MIDI Setup are global. They can be used by all Global Racks and copied into Arrangements, Sketches or Libraries.



# **Saving Devices**

Synfire automatically saves pending changes of all devices to disk. With the **Audio/MIDI Setup** window, you can also save a device manually with *File > Save As* ... to a different place.

# **Device Browser**

This browser helps you organize all your **Device Descriptions** in folders. Global devices are maintained by the **Global Rack**. The folder structure is mirrored in the file system, so every device is an actual file on disk.

The devices that were copied into an **Arrangement Rack** however are stored internally with the arrangement. You won't find them on your hard disk as separate files.

### **Related information**

Devices (on page 36)

# **Device Properties**

Properties Variants, Channels Programs A: Strings EWQL Orch Lean B: Brass C: Woodwinds EWQLOrchestra1Le 0 合 3 variants, 32 sounds on fixed channels, 316 articulations, 100% edited, East West Play (AU) Effect A lean version of the EWQL Orchestra that uses less RAM and Generic contains fewer articulations. 🗸 Send default Volume, Pan, Reverb on Sound Selection Reset MIDI Controllers on Sound Selection -Send Repeated Articulations Allow Modification In Arrange View Controllers Articulations Play (AU) Play (VST)

On the tab Properties you can edit general settings of a device.

#### **Device Name**

The name of the device is used to display it on the list and elsewhere in the software. You can change this name at any time.

#### Manufacturer

This designation should contain a short name for the manufacturer, e.g. **NI** for Native Instruments, or **YAMAHA**. Please always use exactly the same name for the same manufacturer.

#### Model (Plug-in)

Name of the hardware model or plug-in: JV2080, MU90R, EWQLPlay, Kontakt6, etc.

#### Version (Instance)

In case it's a plug-in, put a designation for the loaded preset here that is described by the device description: My Orchestra Kit, Or Misc Basses. You can leave this empty for hardware synthesizers.

#### Сору

This number is incremented every time a device is copied or cloned. It is for your information only, so you can determine which of multiple copies of a device is probably the most current.
# **I** CAUTION:

Manufacturer, Plug-in/Model and Version are used to create a unique file name the device is saved to. Where possible, you should not change this identifier anymore after you began using it in your projects. In order to change the identifier, you need to unlock the input fields first. Synfire will propagate the change to all currently open files to ensure that instruments still point to the same device.

#### Latency

Enter a value in milliseconds by which MIDI data shall be sent ahead of time in order to compensate for the device's latency.

#### Template

Check this box, if this device description is supposed to be a Device Template (on page 37).

#### Effect

Check this box if the plug-in or hardware is an effects processor.

#### Generic

Indicates this device was created automatically when one was needed, in order to serve as a starting point. You may remove this box and edit the device to fully suit your needs.

#### Send Default Volume, Pan, Reverb on Sound Selection

Whether to automatically send <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u>, <u>Reverb</u> and other mixingrelated CC whenever a sound is selected. Disable this if Synfire shall not interfere with a mix you manually set up in your DAW or with external hardware.

#### **Reset MIDI Controllers On Sound Selection**

Some devices require that MIDI CC be reset to default values after a new program/patch was selected. If you notice that your device is behaving strangely with respect to sustain pedals, pitch bend or modulation, you may need to check this option.

#### **Re-Send Repeated Articulations**

Send <u>Articulations (on page 201)</u> every time, regardless whether the previously sent articulation has not changed.

#### **Allow Modification In Arrangement**

Whether you want the **Instrument Inspector** of an arrangement to show the full details of the sound you selected for an instrument. Although this spares you a visit to the device editor on the **Sounds** page, it adds a lot of information that might distract you. This option is always disabled for global (shared) devices.

#### **Custom Controllers, Articulations**

Edit the <u>Custom Controllers (on page 199)</u> or <u>Custom Articulations (on page 201)</u> available for this device.

# **Updating an Embedded Device**

If a device was originally copied into the arrangement, you can update it to the current version of the <u>Global Device</u> <u>Description (on page 39)</u>. This imports a new copy into the arrangement.

# **Device Variants & Channels**

The MIDI ports and channels of a **Device** are configured on the **Variants, Channels** tab. One device may support more than one port and channel layout, called **Variants**.

(	Properties	Variants, Channels	Programs			
EWQL Orch Lean:A Strings		~				
Device not currently used b	y global rack			Fixed So	bund	
1 🗸 CBS KS Master		😂 🖂 Contrabass	✓ ⊗	Sound Name		$\sim$
2 🗸 VCS KS Master		😂 🗹 Cello	✓ ⊗	18V KS M	aster	
3 🗸 VAS KS Master		😂 🖂 Viola	✓ ⊗	Violin		~
4 🗸 11V KS Master		😂 🗹 Violin	✓ ⊗	Method of Sel	ection MSB	LSB
5 🗸 18V KS Master		Violin	✓ ⊗	Program On	ily 🗸 O	0
6 🗸 SVC KS Master		😂 🗹 Cello	✓ ⊗	Program		
7 🗸 SCB KS Master		😂 🖂 Contrabass	✓ ⊗	None	✓ GN	1
8 🗸 SVA KS Master		Series Viola	✓ ⊗	Contro	ollers	
9 🗸 SVL KS Master		😂 🗹 Violin	✓ ⊗	Articula	ations	
$\pm$						
				90	50	
				30	50	
				assigned to t	his channel.	
				>		

For most devices a single port with 16 MIDI channels is sufficient. For comprehensive sound libraries, you may want to add up to four **Variants** A, B, C, D. Each variant represents a different configuration, operation mode, or preset of the same device and supports up to 16 MIDI channels. The advantage of using variants versus creating multiple devices is that **<u>Custom Controllers</u>** (*on page 199*), **<u>Articulations</u>** (*on page 201*) and <u>device properties</u> (*on page 135*) need only be defined once for all variants.

Every MIDI channel can be set to one of three types:

#### **Fixed Sound**

A single sound is permanently addressed on this channel. Its properties can be set using the inspector on the sidebar.

### **Dynamic Program Selection**

An arbitrary program or patch of the device is selected by sending a MIDI bank and program selection message. The list of available sounds is managed on the tab <u>Programs (on page 138)</u>.

## **Reserved Channel**

On this channel only those sounds can be selected that have the same channel number set as their **reserved channel** (for example, this is the case for GM drum kits that require channel 10). These sounds are also managed on the tab **Programs** (on page 138).

Multiple channel types can be mixed in the same variant. If a device has multiple variants with dynamic channels, Synfire considers all of them for sound assignments equally.

A green indicator labeled Used indicates whether a channel is currently in use by one or more instruments, which are listed on the sidebar for your information.

# i Tip:

You can use this page to edit a sound's properties, controllers and articulations while the plug-in is loaded and then call *Copy Sound To User Bank* from the options menu to save it permanently to the **Programs** page. This is handy when you have many sounds to add for a large library.

# i Tip:

For General MIDI (GM) synths and sound generators, we recommend you always exclude channel 10 from dynamic allocation. The standard requires it be reserved for drum kits.



## Note:

Dynamic channels are managed by Synfire automatically. When multiple arrangements are open at the same time, the current foreground window may temporarily put the sounds of a background window aside in order to ensure all sounds of the foreground window are ready to play.

#### **Related information**

Dynamic Sound Allocation (on page 42)

# **Device Programs**

Programs or patches that are selected by sending a MIDI message are managed on the tab **Programs**. Sounds are organized in Banks and classified by **Category** (on page 34).

	Properties	ariants, Channe	s Programs				
		So	oundcase Flu	uidR3			
Banks: $-\pm \bigtriangledown$	Categories:	🖂 Sou			$\square \oplus \boxtimes$		
Bank 000 GM Bank 128 Drumkits User Additions User Additions User Additions	<ul> <li>Instrument</li> <li>Bass</li> <li>Brass</li> <li>Drums</li> <li>Effects</li> <li>Guitar</li> <li>Organ</li> <li>Other</li> <li>Percussion</li> </ul>	Bri Ya	ght Yamaha Grand maha Grand Piano	_	-	Sound Properties Sound Name Yamaha Grand Piano Category Acoustic Piano Preferred Channel Auto	s
Bank Name Bank 000 GM	<ul> <li>Piano</li> <li>Acoustic Pi</li> <li>Clavinet+H</li> <li>Electric Pia</li> </ul>	ano arpsichor no				0:0 [1] Acoustic Graver Controllers Articulations	GM
Method of Selection MSB + LSB MSB 0 LSB 0 V GM	<ul> <li>▼ Strings Cello</li> <li>Contrabass</li> <li>Pizzicato</li> <li>String Enses</li> <li>Synth Strin</li> <li>Viola</li> <li>Viola</li> <li>Violin</li> <li>▶ Synth</li> <li>Voice</li> </ul>	mble js				Yelecity         90         Legend:         A = Articulations         C = Custom CCs         B = Custom Bank Select         * = Missing Category	Volume
.≡ = =×ö" ©	• ⊷ ⊡						
<					>		

## **Sound Banks**

The MIDI standard allows a maximum of 128 sounds per bank. Sound banks are most common for hardware synthesizers and their software emulations. If your device doesn't support sound banks, create a single bank that takes all sounds.

Use the + and - buttons of the banks list to add or remove banks.

#### Name

Be sure to name each bank.

#### **Method of Bank Selection**

For audio plug-ins, the method **Plug-in Preset Selection** usually does the trick. Please consult the MIDI documentation of your device about the MIDI messages sent to select a particular bank and patch/program. Pick that type of message from the drop-down menu. The variables **MSB** and **LSB** are placeholders filled in by the particular sound.

#### **GM (General MIDI)**

Check this box if all sounds in the bank conform to the **General MIDI Standard** (GM). The standard defines a set of program change numbers, names and categories, sparing you the effort of setting up the sounds yourself. Use the list's options menu (right-click) *Initialize General MIDI* to populate an entire bank with sounds according to the standard. Already existing programs will be preserved.

# Categories

This tree shows a hierarchy of Instrument Categories (on page 34). They are basically like folders.

Drop one or more sounds on a category to assign that category to them.

With *Option > Preview Phrase* on the list, you can cut, copy or paste any <u>Phrase (on page 17)</u> you want to use to preview sounds of that category.

### Sounds

The list of sounds can be sorted by **Name**, **Program**, **Category**. You may select one or more sounds to edit their properties on the sidebar inspector.

- Select one or more sounds to edit them as a group together.
- Drag one or more sounds to a different Sound Bank to move them. Hold down 📉 Ctrl to copy them.
- Use keyboard commands Cut, Copy, Paste to copy or move one or more sounds to and from other **Devices** or **Sound Banks**.

## **Creating New Sounds**

Do the following to add a new sound to the list from scratch.

- 1. Select the bank it should be added to.
- 2. Select an appropriate category in the tree.
- 3. Click on + in the list toolbar.

Now fill in the form to configure the sound.

# **Sound Properties**

Edit the properties (meta-data) of a <u>Sound (on page 37)</u> on the <u>Parameter Inspector (on page 54)</u> or the Sounds page.



# $\otimes$

Browse **<u>Template Devices** (on page 37) for already known sound properties in order to copy them into this sound.</u>

#### Sound Name

The name that shows up next to your **Instrument** (on page 33) and when you browse for sounds.

#### Category

Pick a <u>Category (on page 34)</u> from the menu, or drop one or more sounds on the desired category found on the tree.

#### **Method of Selection**

How the program/patch represented by this sound is <u>selected (on page 138)</u> (optional).

#### **Preferred Channel**

Set only if the sound requires or prefers a specific channel. If this is a reserved channel, also tick the box **Reserved**. If the sound is always found on the same channel, you must edit the channel on the **Variants, Channels** (on page 137) tab, not here!

#### Program

The number used to select the sound with a MIDI program change message (1 - 128). Check the GM box, if program change and instrument category conform to the **General MIDI** standard.

### **Controllers, Articulations**

Setup optional <u>**Custom Controllers** (on page 199)</u> and <u>**Articulations** (on page 201)</u> for the sound. Those defined for the device already need not be redefined.

#### Velocity, Volume

These are optional preferences used only for previewing phrases when no such values are available otherwise.

#### Latency

Sound latency (milliseconds). Added to the device latency.

# **Additional Information**

Explains why a sound is not currently audible. Be aware that if you edit a device description, it is not necessarily "online", i.e. currently used by a **Rack Module**. If so, you can't hear the sounds you are editing, of course.

## **Sound Details In Arrangement**

Some properties of a sound can be made visible and edited on the **Parameter Inspector**. This form is usually readonly (grayed out) unless you enabled **Allow For Modification in the Arrange View** with the device description. Device descriptions created by the **Sound Wizard** for convenience do allow this by default, so you can easily alter a sound's properties without visiting the device editor.



# [P

Jump to the device editor where you can edit the sound and all other sounds of the device in context.

#### **Related information**

Instrument Properties (on page 55)

# Parameter Block

This scrollable area shows a <u>Parameter Outlet (on page 45)</u> for all available parameters. It is used to select the current parameter that shows up on the <u>Parameter Inspector (on page 54)</u> to the right of it.



# 0

When this link is active, the **Parameter Views** on the <u>**Track Sheet** (on page 98)</u> show the parameter currently selected on this block. Otherwise they always show <u>**Figure** (on page 294)</u>, so you can edit other parameters with the <u>**Parameter Inspector**</u> (on page 54)</u> without losing sight of the figures.

Edit

Double-click on an outlet to open or close the <u>Phrase Editor (on page 86)</u> or <u>Progression (on page 105)</u> page.

### **Drag & Drop**

Parameter data can be <u>copied or moved to and from other outlets (on page 242)</u> with drag & drop or copy & paste. Parameter data of a different type is <u>converted automatically (on page 324)</u> to fit its destination. Hold down <u>C</u> Ctrl while you drag in order to copy a parameter.

### Delete

Delete parameter data with the Delete key.

### **Right-Click Menu**

Right-click on an outlet to open the Parameter menu with various editing options.

# Scrolling

Scroll the block up or down with the mouse wheel to make hidden outlets visible.

### Groups

Collapse or expand individual parameter groups to make room or focus on a particular group.

# **Parameter View**

Shows the selected parameter of the selected phrase. Unless you linked it permanently to the <u>Parameter Block (on</u> page 143), it shows Figure (on page 294) by default.

Parameter views with slightly different layouts are found on the <u>Track Sheet (on page 98)</u>, the **Overview** page, the embedded <u>Phrase Editor (on page 86)</u> and the <u>Phrase Pool Editor (on page 123)</u> of a library.



### ♥ Open

Full view of the parameter. You can select and edit data right away, or double-click to open the more detailed **Phrase Editor** (on page 86).

# Collapsed

Read-only summary of the parameter, leaving more space for other instruments.

### Minimized

The entire track shrunk to a label, it is merely a placeholder for the instrument, leaving maximum space for other instruments. Double-click to open, if you want to edit data. Note that phrases with existing data can't be minimized.

Switch between three presentation modes. Hold down **1** Shift to apply the change to all containers equally.

- 1. **Open**: Phrases with at least one parameter show a full-size **Parameter View** (on page 144) and others are minimized.
- 2. **Collapsed**: All tracks are collapsed to a summary of the same size.
- 3. **Minimized**: Instruments without a phrase are minimized to make room. Phrases with a phrase show a summary. This is the most compact presentation.

#### **Related information**

Parameter Block (on page 143) Phrase Editor (on page 86)

# **Display Options**

On the View and Playback menus, you have several options to customize what is displayed on screen.

### **View Menu**

#### **Parameter Trace**

Indicates with green rulers in the **Structure View** which containers provide the currently selected parameter for the currently selected instrument. This helps you quickly spot all places where a parameter takes effect and where it is (temporarily) overshadowed by another container.

#### **Parameter Hints**

Adds explanatory text to the track sheet that helps understand where a parameter comes from, why it possibly doesn't show up, or why it looks the way it does.

#### **Global Parameter Shadow**

Adds a dimmed display of inherited parameter data to the background, so you understand what is happening even when the parameter is not physically present in the current container. This slows down the display a bit but can be very helpful.

#### **Looped Parameter Values**

Rolls out looped data until the end of the current container, so you can see how it unfolds along other parameters and the grid. This is always enabled for **Figure**, because segments often overlap and you need to know how the next loop merges into the previous one.

#### **Centered Symbols**

Makes **Figure** symbols look more like square note heads rather than flat MIDI notes. If you are used to working with notation programs, this might be more intuitive. Disable this when you need to position and quantize symbols with maximum precision.

#### **Articulations**

Adds the names of articulations to a Figure display.

#### Grid

Shows the currently selected grid as an overlay on a **Parameter View**. This is useful when you want to understand the implications of combined or irregular quantization grids.

#### **Playback Menu**

#### **Scroll Views With Playhead**

Scroll views such that they are following the current playback position.

#### **Follow Parameter Trace**

Select the container that provides the currently selected parameter for the currently selected instrument at the moment of playback.

# Transport

Transport commands can be customized with *Preferences > User Interface > Edit Keyboard Shortcuts* and configured with *Playback > Audio/MIDI Setup > MIDI > Remote Transport Control*.



### **Transport Bar**

### И

Rewind: Reset playhead to the very beginning.

### 

Stop: Stop all playback.

# 

Pause: Halt playback at current position. Click again to continue playing from there.

# $\triangleright$

Play: Start or restart playback (Space). If something is selected on the Time Ruler or in a <u>Parameter</u> <u>View (on page 144)</u>, playback starts from there. Hold down <u>Alt</u> to play only instruments with phrases in the current container.

# R

**Solo**: Start playback, rendering only the currently selected tracks. This is different from soloing the audio signal with the **S** button.

# 60

**Preview**: Listen to the currently selected object by rendering an example phrase. You can also use this to preview **Harmony** or any other parameter that's currently selected. Enable *Playback > Tie Auto-Chord Notes* to sustain all notes that are already playing in the previous chord.

# С

Record: Record the currently selected phrase from external MIDI input. See: Recording (on page 260).

#### ~

**Record Parameter**: Record anything from external MIDI input and convert it to the currently selected Parameter. See: <u>Recording Parameters (on page 261)</u>.

# 0

Loop: Run next playback in cycles as marked on the Time Ruler.

# $\bigcirc$

**External Sync**: Transmit TEMPO, LOCATE, START, STOP commands to external hardware in synchronization with local playback. See: <u>External Synchronization (on page 66)</u>.

# 0

Overdub: Don't clear the phrase before recording, i.e. add whatever is recorded to the existing phrase.

**Bounce to Disk**: Save audio output of next playback to a file. The file is placed next to the arrangement document (*File > Show File in Finder*). Requires all instruments to use the same **Audio Engine**.

Select the root container to create an audio file for the entire arrangement. You may optionally select a starting position on the timeline. Recording will stop at the end of the root container, so it is a good idea to leave have a little room at the end for all sounds to settle.

Select one or more child containers to trim the captured audio file exactly to their bounds. Stop playback after all containers have been played.

Press Play to begin the recording and Stop to end it.

# Sketch App

# Editing real-time sketches

This is basically a very limited arrangement window for editing a <u>Sketch (on page 32)</u>.

File Edit I	nstrument P	hrase Parar	meter Playb	ack Windov	v Help			
$\langle \rangle$	Ø		C		2 @ O	ОК	Cancel	) 🔆 🎘
	Matrix	Sounds						
		D	ancing Elepha	ant				
Sketch Scheme	<ul><li>■ Take</li></ul>	2 Take	3 Take	<ul><li>▲</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li><li>♦</li>&lt;</ul>			Instrument Label Rhodes	7
<ul> <li>Harmony</li> <li>Trans.</li> <li>Interpr.</li> </ul>	<ul> <li>Figure</li> <li>Rhythm</li> <li>Transp.</li> </ul>	Figure Rhythm Transp.	Figure Rhythm Transp.	Figure Rhythm Transp.			Global Rack	nges
Reverb Pan Volume	Variation	Variation	Variation	Variation			Soundcase FluidR3 03 Legend EP 2	<ul> <li>✓ </li> <li>✓ </li> <li>✓ </li> <li>↔ </li></ul>
	Pan Volume	Pan Volume	Pan Volume	Pan Volume			Soundcase AU	
Master	<i>S</i> M <i>S</i> M <i>S</i> → <i>S</i> → <i></i>		s × · · · · · · · · · · · · · · · · · ·	<i>₽</i> s M <i>*</i> * *			Controllers A ▷ Sound Details	
	Rhodes	Yamaha Grand Piano	Pad	Bass				

# Note:

This feature is available with the **Express** and **Pro** editions.

# **Standalone Editor**

Open a new Sketch App with File > New > Sketch or open any existing Sketch with File > Open ....

## **Editing The Current Sketch**

With *Playback > Edit Current Sketch* a Sketch App opens on the currently active real-time Sketch in dialog mode. Pressing the button **OK** saves the Sketch back to the *Playback* menu.

### Adding Sketches To The Playback Menu

Save a Sketch with *File > Save As* ... to the Sketches folder inside the Config folder. It will automatically appear in the menu after restarting Synfire.

### **Converting a Sketch To an Arrangement**

If you happen to like a particular Sketch and want to use as a starting point to build an arrangement, you can convert it with *File > Convert > Open As Arrangement*.

### **Related information**

Sketches (on page 32)

# Song App

# Legacy window from Harmony Navigator 2

This app was retained in order to support migration of files save with **Harmony Navigator 2 (Advanced Edition)**. You should no longer use it and start new projects with the **Arrangement App** instead.

# Library App

# The standalone editor for libraries

Although the <u>Arrange Window (on page 91)</u> allows for editing any <u>Library (on page 28)</u>, embedded or otherwise, you may rather want to use this standalone app to work on one or more libraries simultaneously.



This app is designed to be a workbench for importing, generating, editing and optimizing large collections of phrases, the lifeblood of **Music Prototyping**.



This feature is available with the **Express** and **Pro** editions.

# Browser

On the left sidebar, you can search, browse and organize the **Phrase Pools** in a library.

Open multiple instances of the Library App to move data around with drag & drop or copy & paste.

In its collapsed state with the **Phrase Editor** closed, the app can be tacked to the desktop to stay always on top, which eases data exchange between multiple windows.

### **Phrases**

A grid with all phrases in the selected pool shows at the top.

- Select a phrase to load in to the editor below.
- Drag a phrase to another pool, another library window, or into an arrangement.
- Drop a phrase from anywhere to add it to the pool.
- Cut, Copy, Paste, Delete or Duplicate a phrase with the respective keyboard shortcuts.

# **Phrase Editor**

The bottom right section is a regular **Phrase Editor** for editing the currently selected phrase in the pool. Four pages **Editor**, **Harmonizer**, **Figure Recognition** and **Factory** are dedicated to specific tasks that are involved with this.

Learn more about the Phrase Editor (on page 123) here.

## **Related information**

Libraries (on page 28)

# Palette App

# The standalone editor and panel for multi-palette surfing

### Video Tutorial

Although a **Palette** page is integrated with an arrangement window, you may want to open additional palettes to explore the relationships of multiple keys, or compose a key change.

File Edit Palett	te Layout Colo	ring View Navig	ation Playback	Window Help					
1 2 Cmaj9 F 	Fmaj7 Ar -	n 14	<b>G6</b> (ac	id9)					
< > # 2					G7sus4(10) Gsus2				
					7sus2 9sus4				
😑 Harmony	Cmaj7			Fsus2(b5)	G				
Freeze MIDI Input MIDI Live Tempo	maj/(add)    maj/(13)    maj/(13)    maj/(13)    f(add)    f(add)    f(add)    f(add)    (add)    add)    ↓	Daus4 75us4 9sus4 Daus2 7sus2 Dm m6 m7 m6(add9) m7sus4 m6(add9) m11 m11(#9) m13 m13 m13 m13 m13 m13 m13	Esus4 7sus4 Em m7 m(b9) m7sus4 m7(s9) m1(#9) Em (#5) m7(#5) m1(#5)	F S (add2) (add2) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add9) (add2	2 (add2) (add4) 6 7 (add9) 6 (add4) 6 (add4) 9 7(13) 7(9,13) 73 7(9,13) 73 73 79,11,13) Csus4 V	W	Asus4 7sus4 9sus4 Asus2 Tous2 Am(#5) m7(#5) m709 m709 m71 m1 m1 m1 m1 m1 m1 m1 m1 m1 m		Bm(#5) m7(#5) Bdim m7(b5) VII
melodic-major	s					5		ЬVII	
	Caug C7 9 C7(45) maj1(45) C7us2 C7us2 C7us4 7us4(10)	Ddim mr7(b5) Daus2(b5) Ddim7 Dm(#5) m7(#5)	Eaug 7(#5) 7(#5,#9) Edim 7(b5) 7(b5) 7(b7) 7(#9) 7(b9,#9)	Fm s mf(add9) m(add9) fd(add9) Found Fsuud	Gm d m7 m(b9) m7544 m7(b9) m11(48) G7(b9)	Ab(65) (6(05) maj7(45) Abaug maj7(45) Abdug (m/(45) Abaus2(05)		Bb(b5) G(b5) 705) Bb (add2) G(add2) G(add2) G(add2) 9 7(3) 7(3,411) 7(9,411) 7(9,413) 2050 Bbsus2 7052 7052 7052 7055 7057 7	Bdim7

Open a new standalone Palette window with File > New > Palette, or double-click on a key in the Circle Of Fifths.

# Note:

This feature is available with the Express and Pro editions.

Synfire comes with a collection of example palettes for various purposes. You find them with File > Open Examples ....

# 平

Tack the window to the desktop to always stay on top of other windows. This is handy if you want to drag chords to another window.

# Ø

Open the **Notepad**, which is a small <u>Harmony Lane (on page 97)</u> for collecting chords. This lane is however not linked to any arrangement.

#### Harmony (Outlet)

Drag the Notepad progression off to any destination, or drop a progression from elsewhere.

# Ο

Record the chords you play on all palettes into the Notepad.

### **Related information**

Playing Palettes (on page 240) Editing Palettes (on page 108) Palette Layout & Coloring (on page 109)

# **Progression App**

# The standalone editor for one or more progressions

# Video Tutorial

Outside of an Arrangement or Library, **Progressions** (on page 14) can be edited with the standalone **Progression App**, too. Open an empty app with *File > New > Progression*, or open a file with the .cogpro extension.

You may want to exchange progression files with other users. Larger collections of progressions however are best maintained in a Library (on page 28).



# Note:

This feature is available with the Express and Pro editions.

#### **Related information**

Editing Harmony (Progressions) (on page 250) Controlling Inversion, Voicing and Octave (on page 254) Progressions (on page 14)

# Catalog App

Edit the fundamental interval structures Synfire is working on

Browse and edit the current <u>Catalog (on page 16)</u> with Edit > Catalog of Chords And Scales.



Double-click on a scale or Scale Set to open a new palette on them. You may deactivate or delete chords or scales that you don't want or need. Some basic structures however can't be deleted or modified.

The **interval structures** listed in the Catalog are independent of root pitch. A root pitch is added only when a particular instance is created. For example, the structure m9 is used by instances of Am9, Gm9, Ebm9, etc and maj7 is used by instances of Amaj7, Bbmaj7, etc.

# **Creating And Modifying Items**

Before you begin, please make sure you understand how the Catalog (on page 16) influences the behavior of Synfire.

As soon as you begin editing an item's intervals, a copy is put aside (highlighted in the list with green color) that you can continue to modify until you commit all changes with the Save button, or click elsewhere to discard them.

Alternatively you can begin editing a copy with *Item > New Clone* .... If you want to start a new item from scratch, create an empty item with *Item > New*.



# CAUTION:

All open documents, palettes and progressions update their contents automatically according to the changes you make to the Catalog. Thus, if you want to make sure your edits don't meddle with other files, you should close them before you edit the Catalog.

# **Status Colors**

Items may show up in different colors that indicate a certain status.

#### Yellow

Your custom-defined catalog entries.

#### Red

The name or alternative names have conflicts with already existing items.

#### Italic

Temporary items not persistently saved with the catalog.

#### **Item Editor**

#### Name

Desired name of the interval structure *without* a root pitch. Make sure you understand the <u>naming</u> <u>conventions (on page 325)</u>. If you have already set all intervals, you can get a suggestion by looking into the menu *Item > Rename* ....

#### Suggestion

Click this button to pick from a list of suggested names.

#### **Alternative Names**

A list of alternative names separated by spaces. Synfire recognizes these as synonymous with the name when it parses your input on the progression editor.

#### **Priority**

Influences scale selection and other internal deliberations where multiple results need to be narrowed down. You can also use the slider to change this value. Use this setting if you want an item to be used more often than another equivalent one. Zero is top priority and nine is for items you want to avoid.

#### **Persistent (Save)**

Check this if you want to permanently save the item with the catalog. If disabled, it will be discarded when no longer used.

#### Active

Disable this if you want to exclude the item from palettes and all harmony calculations. You can use this to hide an item without deleting it.

#### Recognize

Whether Synfire shall consider this item as a possible candidate for harmonization. You can disable this for chords you don't want to see in the results of the Harmonizer. It should remain disabled for very large interval structures that are too ambiguous to be meaningful as a chord.

#### **Use as Extension**

Whether a chord should be considered an extended version of a more simple chord when computing the **Jazz Up** feature. You can exclude some extremely dissonant chords here that should not be used unless you place them somewhere deliberately.

#### **Interval Structure**

You can add or remove intervals by ticking the boxes, or by holding down  $\sim$  Alt while you click on the MIDI keyboard widget.

While chord structures may span two octaves, scales are only one octave wide.

Rotate intervals around with Invert + and Invert -

## Note:

Items in the Catalog are pitch-agnostic interval structures written without a preceding note name. A root note is added only when a chord or scale is created from the structure. For example, interval structure **m9** may be used to create the chords **Em9** or **Gm9** and structure **maj7** may be used for **Amaj7** or **Bbmaj7**.

### **Cross References**

Equivalents and Extensions lists structures that are built from the same intervals but bear another name.

**Rotations/Modes** lists structures that consist of the same pitch classes but have a different interval structure. That is, these structures are equivalent but start at a different root.

#### **Saving Changes**

Changes immediately propagate to all open documents and windows, updating palettes and progressions as you edit.

Don't forget to save your changes with the button Save. This makes sure the entire catalog is re-calculated to accommodate the modified item. It is also a good idea to save the Catalog file with *File > Save* from time to time, although it will also be saved when you close the window.

#### Note:

If you want a customized Catalog to be opened and used when Synfire starts, you must save it to the **Config** folder using the name **Tonality.catalog**. You can restore the factory default with *File > New*.

### Note:

Note that the chord structure **Major** has an empty name because a major chord is denoted by its uppercase root note only. Make sure you understand the <u>naming conventions (on page 325)</u>.

# Note:

With *Item > Show Temporary Items* you can show chords, scales and scale sets that Synfire generated on the fly for internal calculations.

Note:

This feature is available with the Express and Pro editions.

#### **Related information**

Harmony Concept (on page 6)

# Audio/MIDI Setup App

# Configure basic audio communication and edit the current global rack

Manage your audio settings, the <u>Global Rack (on page 39</u>), global <u>Device Descriptions (on page 36</u>) and your <u>Global</u> <u>Instruments (on page 40)</u> in this window. It can be opened from any window, either from the toolbar or via *Playback* > *Audio/MIDI Setup*.



# Toolbar



Reset the Audio/MIDI system. This may help return to a more stable state after things went wrong.

# $\bigcirc$

This MIDI input indicator flashes when incoming notes are detected (Controller messages are not currently indicated).

# ?

Open the Embedded Help browser.

#### **Related information**

Audio System (on page 32)

# **Audio Settings Page**

Set your audio output device, sample rate and block size here. You are probably familiar with these terms from your DAW already. Please follow the tool tips for more information on a particular switch.



## **Selecting an Engine**

Select from the menu button the **Audio Engine** or **Embedded GM Synth** the settings of which you want to adjust. You have the option of running multiple engines on different computers in your LAN.

### **Granting File Access**

On macOS, audio plug-ins loaded by the **Audio Engine** may not immediately be able to access an external volume or folder where you are keeping their resources and sounds. If individual plug-ins don't load presets, contents or sounds

as expected, you may need to grant the Audio Engine access permission to the volume or folder where the sounds are stored.

Click on **Grant File Access** to open a file dialog where you open the volume or folder that includes those sounds. This will make macOS grant the Audio Engine access permissions to that location.

# **Scanning Installed Plug-ins**

Pressing on Scan Installed Plug-ins will search the hard drive of the selected engine for installed audio plug-ins. When the subsequent dialog pops up, be sure to include all directories containing VST plug-ins on your computer before you run the scan.

Select what kind of scan you want to run.

- 1. **Update or Continue**: Looks for plug-ins that have changed since the last time a scan was run. If the last scan did not run to completion, this will continue the scan where it was interrupted. This is what Synfire does on start-up.
- 2. Scan All: Rebuild the entire list of plug-ins. Any plug-in that was put aside during a previous run because it is broken or not working will be ignored.
- 3. Scan All (Reset): Rebuild the entire list of plug-ins and also reset the list of broken plug-ins. This will look at every plug-in and try to make it work again.

A scan is usually relatively fast if you have no more than 500 plug-ins installed on your computer. For 1,000 or 2,000 or more plug-ins, it can take quite a while. If you need to interrupt a very long scan, you can resume it at a later time.

# Note:

On Microsoft Windows, you may experience problems when multiple programs attempt to access the same ASIO driver. In order to be able to run the engines and a DAW in parallel, you might need to assign them to different audio output drivers. For example, you may want to use the ASIO driver for your DAW and the Windows Audio or DirectX drivers for the engines. On the Mac, running multiple Engines and DAWs at the same time is not a problem.

# Preferences

#### **Enable Built-in GM Synth**

Synfire comes with a built-in sample player that offers a selection of GM compatible sounds. You should enable this only if you can't run the **Audio Engine** on your computer for some reason. Hosting the **Soundcase** plug-in is equivalent to using the built-in GM synth, however with less impact on performance.

#### **Enable Audio Engine**

You have the option to run Synfire without the **Audio Engine**, based on MIDI and external hardware alone. We do not recommend this, though.

#### Launch Immediately on Start-up

Disabling this will cause the **Audio Engine** to be launched only when it is first used. This does not save much time, but may be useful when working with simple MIDI projects.

#### **Keep Running After Quit**

If enabled, Synfire will leave **Audio Engine** running after it quits. When you restart Synfire again, the global rack is already loaded, which can save you time. We do not recommend this.

#### **Fast Loading**

Synfire avoids loading a preset, if a plug-in already has the same preset loaded. This can save load time, especially for large sound libraries. Disable this if you encounter any issues with loading plug-in presets.

#### **Open Plug-in Editor After Loading**

Disable this if you find it annoying. A few plug-ins also crash if their editor is opened too quickly after the plug-in was loaded.

#### Ask Before Saving Global Rack

Synfire quietly saves pending changes to disk. If you want to get notified and get a chance to decline, you can enable this option.

#### Load Last Global Rack on Start-up

Synfire will load the last used Global Rack on start-up. Disable this if you want otherwise.

#### Load Global Racks Associated With Arrangements

Arrangements remember the Global Rack that was open when they were last saved and restore it when they are opened again. This is only a global default. You can enable or disable this link in each arrangement individually in the toolbar on its **Sounds Page**.

# **MIDI Settings Page**

Setup dynamic sound allocation, keyboard split, metronome and remote control on this page. This configuration is saved as part of the **Global Rack** file.

File Edit View Playback Window Help			
< >	📰 Global Rack 🛛 📾 Global Instruments	🖉 Global Device Descriptions 🖾 Repository	5 ® O ?
Sound Management	Keyboard Split split: C3 →]	Remote Transport Control           Rewind:         Ctrl:112 Ch:1 Val:127         →]	
Swap Programs when Low on MIDI Channels		<b>Stop:</b> Ctrl:114 Ch:1 Val:127 →]	
Reset Controllers on Sound Selection	Metronome: Click 1: $C#3 \rightarrow$ Click 2: $D3 \rightarrow$ Measures: $1 \sim 6$ b Disable	Pause:       →]         Piay:       Ctrl:115 Ch:1 Val:127       →]         Solo:       →]         Probe:       →]         Record:       Ctrl:117 Ch:1 Val:127       →]         Rec. Param:       →]	

### **Sound Management**

#### Instrument Accepts Playing Ranges From Sound

With this option enabled, Synfire will copy the **Playing Ranges** of a **Sound** to an **Instrument** when you assign one. Be aware though that this will re-render current MIDI output to match the new ranges. You might want disable this, if you are concerned with preserving MIDI current output.

#### **Swap Programs When Low On MIDI Channels**

If more sounds are requested than there are channels available, unused sounds will be temporarily swapped out (default setting). This is especially important when you open multiple arrangements at the same time.

#### **Also For Global Instruments**

Makes above automatism also apply to **Global Instruments**. We recommend this, unless you have a dedicated plug-in or sound module reserved for Global Instruments exclusively.

#### **Reset Controllers on Sound Selection**

Sends a Reset-All-Controllers MIDI message after selecting a new sound. This global default can be overridden for each device.

# i Tip:

If you ever have the feeling that Synfire has messed with your sound assignments, press **Reset** on the window toolbar. This will flush dynamic allocation and request all sounds anew.

# **Keyboard Split & Metronome**

#### **Keyboard Split**

When this is set, only keys below the split are considered for <u>Live Chord Detection (on page 262)</u> and higher notes can be played freely as an accompanying melody. Delete this to consider all notes for chord detection.

# →]

Click to learn a pitch from MIDI input. Let the listening time out to delete the pitch.

#### Click 1 & 2

Enter or learn the notes to be played for the metronome.

#### Lead-in

Number of measures to use for lead-in before a recording starts. Set this to zero, if you don't want the transport to rewind for a pre-roll.

#### Disable

You can disable the metronome altogether.

## **Remote Transport Control**

Here you can configure MIDI notes or CC for start, Stop, Record and other transport commands. These controls only work in the Arrange window.

### **Related information**

Setting Up MIDI Input (on page 233) Live Chord Detection (on page 262)

# **Inputs Page**

All currently known MIDI ports able to receive MIDI data are listed here. This configuration is saved as part of the **Global Rack** file.

File	Edit View Playback \	Vindow Help						
<	> II -ili Aud	io 🕐 MIDI	ightarrow] Inputs	📰 Global Rack	াট্রা Global Instruments	Ø Global Device Descriptions	C Repository	₽® <sup>©</sup> ?
Input	Port IAC Driver Bus 1 MIDI Port	Status	V Lo	Associ opback	ated Device Description(s)		<ul> <li>✓ MIDI Through</li> <li>✓ Verify MIDI Input</li> <li>✓ Compensate Late</li> </ul>	Port on Startup
•	IAC Driver Bus 2 MIDI Port		V Lo	opback			✓ Use Audio Engine	e For MIDI Input
•	IAC Driver Bus 3 MIDI Port		V Lo	opback				
•	IAC Driver Bus 4 MIDI Port		V Lo	opback				
<b>v</b>	Impulse Impulse MIDI Port	Engine		opback				
•	Impulse Impulse MIDI In MIDI Port	Input Only Engine						

# **Select Input Ports**

Select any ports you want to use for recording phrases with Synfire by checking the respective box. It will turn red. Any input received on those ports should make the MIDI connector icon on the top toolbar flicker. Note however, that only incoming note messages are indicated.

If you connected hardware to your computer that does not appear in the list, check your cabling and USB driver setup and press the **Reset** button on the top toolbar to refresh the list.

# i Tip:

The **Reset** button can be used at any time to put the audio and MIDI setup back into a defined state (for example, if you suspect that something got messed up, or if there is no sound at all anymore). Your settings are not be affected by a reset.

### **Loopback Drivers**

So-called **Virtual MIDI Cables**, or loopback drivers, can be used to route the MIDI output of Synfire to other software on your computer, for example a DAW. It also works the other way around. On the Mac, this is the **IAC Driver** made by Apple. For Microsoft Windows, there is a small selection of drivers available, for example **LoopBe** by nerds.de, or **MIDI Yoke**, and others.

Since these drivers bounce back to Synfire all data they just received from Synfire, there is a high risk of an infinite loop to occur, if at the same time **Midi Through** is enabled (Feedback). This infinite loop puts an extreme load on the Audio Engine. To prevent this, please mark such ports as **Loopback**.

# **Control Surfaces**

With the switch **Control** you may tag any input port as a control surface or other MIDI controller that is not supposed to provide input for recording. Incoming messages can be mapped for remote control only.

### **Settings**

# $\bigcirc$

**MIDI Through**: If enabled, incoming MIDI data is forwarded immediately to the instrument that was last selected. This is helpful when you want to play the currently selected instrument with an external MIDI keyboard.

The icon also indicates incoming MIDI notes by flashing.

#### Verify MIDI Input on Start-up

If enabled, Synfire will issue a warning on start-up if it can not find any MIDI ports enabled for input.

#### **Compensate Latency During Recording**

After a recording finished, Synfire analyzes the <u>Take (on page 312)</u> for timing problems and corrects these automatically. You can disable this, if you feel it leads to undesired results.

#### **Use Audio Engine For MIDI Input**

The <u>Audio Engine (on page 41)</u> supports a tighter timing than the user interface application of Synfire. This option makes the Engine receive incoming MIDI data and forward it to Synfire with timestamps already applied. You can disable this, if you experience problems on Windows with sharing USB MIDI drivers among multiple programs.

#### **Snap Input To Harmony**

Snaps all MIDI input to the current vertical scale. This mode is active only during recording, live chord detection and palette playback. If a global **Keyboard Split** is set, only notes greater or equal this split are snapped to scale during <u>Live Chord Detection (on page 262)</u>. Of course all this only makes sense when a Harmony parameter is already present.

#### **Operate Drone in MIDI Mode**

A **<u>Drone** (on page 43)</u> can optionally be operated in MIDI-only mode. If so it will no longer host an audio plug-in, but receive and forward MIDI data only. Consider using a <u>MIDI Drone (on page 44)</u> instead.

# CAUTION:

Under Windows, problems may occur if multiple programs attempt to use the same USB MIDI drivers at the same time. Especially prone to this issue are cheap drivers that often accompany very inexpensive MIDI hardware. If you wonder why Synfire is unable to receive data from your hardware, there is probably another program already using the USB driver. In order for Synfire and other software to receive MIDI at the same time, you may need to use separate USB interfaces, or look for a more capable driver, if possible. Since this is a frequent problem, you should visit the user forum for the latest solutions (users.cognitone.com)

### **Related information**

Setting Up MIDI Input (on page 233)

# **Global Rack Page**

On this page you see the **<u>Global Rack</u>** (on page 39). This rack is hosting sounds that are available to all open arrangements and other files at the same time. Learn more <u>about editing racks</u> (on page 129) here.



### **Related information**

Rack Editor (on page 129)

# Sidebar: Rack Module Library

### Adding sounds made easy

#### Video Tutorial

This browser on the left sidebar shows various items that can be used to create or change a **Rack Module** in order to provide sounds for instruments. Open the browser with **Open** the window toolbar.

Browse and search for items that you can drop on an **Instrument**, a **Rack** or a **Rack Module** in order to create or change a rack module and use its sounds for instruments.

Search	Rack Module Library	
<ul> <li>Embedded GM Synth</li> <li>Rack Module Presets</li> <li>Effects</li> <li>Kontakt</li> <li>Orchestral</li> <li>Physical Modeling</li> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>String Studio VS-2 (AU)</li> <li>String Studio VS-2 (AU)</li> <li>String Studio VS-2 (AU)</li> <li>String Studio VS-2 (AU)</li> <li>Strom GS-2 (AU)</li> <li>Tenor Sax</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> </ul>	Search	$\otimes$
<ul> <li>Rack Module Presets</li> <li>Effects</li> <li>Kontakt</li> <li>Orchestral</li> <li>Physical Modeling</li> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>String Studio VS-2 (AU)</li> <li>String Studio VS-2 (AU)</li> <li>Strim GS-2 (AU)</li> <li>Tenor Sax</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 3</li> <li>Soprano Sax</li> </ul>	🐨 Embedded GM Synth	
<ul> <li>Effects</li> <li>Kontakt</li> <li>Orchestral</li> <li>Physical Modeling</li> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>W Drones (Audio)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	🔻 🖻 Rack Module Presets	
<ul> <li>Kontakt</li> <li>Orchestral</li> <li>Physical Modeling</li> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Obce (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> <li>SWAM Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul>	Effects	
<ul> <li>Orchestral</li> <li>Physical Modeling         <ul> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> </ul> </li> <li>Soprano Sax         <ul> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> </ul> </li> <li>Samplers         <ul> <li>Synthesizers</li> <li>Addio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MID)</li> </ul> </li> <li>External MIDI         <ul> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> </ul> </li> <li>Preset Properties:         <ul> <li>Soprano Sax</li> </ul> </li> <li>SWAM Soprano Sax (AU)</li> </ul> <li>Soprano Sax</li>	🕨 🗀 Kontakt	
<ul> <li>Physical Modeling         <ul> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> </ul> </li> <li>Addio Plug-ins         <ul> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> </ul> </li> <li>Preset Properties:         <ul> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul> </li> </ul>	🕨 📄 Orchestral	
<ul> <li>Clarinet (AU)</li> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>Sprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Strum 6S-2 (AU)</li> <li>Tenor Sax</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> </ul>	🔻 🗀 Physical Modeling	
<ul> <li>Lounge Lizard EP-3</li> <li>Lounge Lizard EP-4</li> <li>Oboe (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (Audio)</li> <li>AW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> <li>SWAM Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul>	🗅 Clarinet (AU)	
<ul> <li>Lounge Lizard EP-4</li> <li>Obce (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul>	🗋 Lounge Lizard EP-3	
<ul> <li>Poboe (AU)</li> <li>Soprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> </ul>	🗅 Lounge Lizard EP-4	
<ul> <li>Noprano Sax</li> <li>Noprano Sax</li> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Somplers</li> <li>Synthesizers</li> <li>Addio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MID)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul>	🕒 Oboe (AU)	
<ul> <li>String Studio VS-2 (AU)</li> <li>Strum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax (AU)</li> <li>Soprano Sax (AU)</li> </ul>	🗋 Soprano Sax	
<ul> <li>Netrum GS-2 (AU)</li> <li>Tenor Sax</li> <li>Samplers</li> <li>Synthesizers</li> <li>Audio Plug-ins</li> <li>DAW Drones (Audio)</li> <li>DAW Drones (MIDI)</li> <li>External MIDI</li> <li>IAC Driver Bus 1</li> <li>IAC Driver Bus 2</li> <li>IAC Driver Bus 3</li> <li>IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	🗋 String Studio VS-2 (A	U)
<ul> <li>▶ Tenor Sax</li> <li>▶ Samplers</li> <li>▶ Synthesizers</li> <li>▶ Audio Plug-ins DAW Drones (Audio) DAW Drones (MIDI)</li> <li>▼ External MIDI</li> <li>♥ IAC Driver Bus 1</li> <li>♥ IAC Driver Bus 2</li> <li>♥ IAC Driver Bus 3</li> <li>♥ IAC Driver Bus 4</li> </ul> Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	🕒 Strum GS-2 (AU)	
<ul> <li>► Samplers</li> <li>► Synthesizers</li> <li>► Audio Plug-ins DAW Drones (Audio) DAW Drones (MIDI)</li> <li>▼ External MIDI</li> <li>④ IAC Driver Bus 1</li> <li>④ IAC Driver Bus 2</li> <li>④ IAC Driver Bus 3</li> <li>④ IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	🗋 Tenor Sax	
<ul> <li>È Synthesizers</li> <li>Audio Plug-ins DAW Drones (Audio) DAW Drones (MIDI)</li> <li>External MIDI</li> <li>Ch C Driver Bus 1</li> <li>AC Driver Bus 2</li> <li>AC Driver Bus 3</li> <li>AC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	Samplers	
<ul> <li>Audio Plug-ins DAW Drones (Audio) DAW Drones (MIDI)</li> <li>External MIDI         <ul> <li>① IAC Driver Bus 1</li> <li>② IAC Driver Bus 2</li> <li>③ IAC Driver Bus 3</li> <li>③ IAC Driver Bus 4</li> </ul> </li> <li>Preset Properties: Soprano Sax</li> <li>SWAM Soprano Sax (AU)</li> <li>Soprano Sax</li> </ul>	Synthesizers	
DAW Drones (Audio) DAW Drones (MIDI) ♥ External MIDI ① IAC Driver Bus 1 ② IAC Driver Bus 2 ② IAC Driver Bus 3 ③ IAC Driver Bus 4 Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	Audio Plug-ins	
<ul> <li>DAW Drones (MIDI)</li> <li>▼ External MIDI</li> <li>① IAC Driver Bus 1</li> <li>① IAC Driver Bus 2</li> <li>① IAC Driver Bus 3</li> <li>① IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	DAW Drones (Audio)	
External MIDI	DAW Drones (MIDI)	
C IAC Driver Bus 1 C IAC Driver Bus 2 IAC Driver Bus 3 IAC Driver Bus 4 Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	▼ External MIDI	
<ul> <li>(*) IAC Driver Bus 2</li> <li>(*) IAC Driver Bus 3</li> <li>(*) IAC Driver Bus 4</li> <li>Preset Properties:</li> <li>Soprano Sax</li> </ul>	() IAC Driver Bus 1	
C IAC Driver Bus 3 C IAC Driver Bus 4 Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	() IAC Driver Bus 2	
ey TAC Driver Bus 4 Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	IAC Driver Bus 3     IAC Driver Bus 4	
Preset Properties: Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	IAC Driver Bus 4	
Soprano Sax SWAM Soprano Sax (AU) Soprano Sax	Preset Properties:	
SWAM Soprano Sax (AU) Soprano Sax	Soprano Sax	
SWAM Soprano Sax (AU) Soprano Sax		
SWAM Soprano Sax (AU) Soprano Sax		
SWAM Soprano Sax (AU) Soprano Sax		
SWAM Soprano Sax (AU) Soprano Sax		
Soprano Sax	SWAM Soprano Sax (AU)	
	Soprano Sax	

#### 

**Embedded GM Synth**: Drag it to an **Instrument** or **Rack** if you want to use a sound of the embedded synthesizer. The advantage being that your arrangement or Sketch will work on any installation of Synfire, not depending on specific plug-ins. This item is grayed out if the embedded synthesizer is disabled.

n

**Rack Module Presets**: All presets you have saved so far will appear here for browsing. Create additional folders and organize your presets as you see fit. In the panel below, you can edit names and comments.

Drag to an Instrument or Rack to use the sounds of this preset.

Drag a preset with an effects plug-in to a **Rack Module** to load the plug-in as the module's insert effect.

Å

**Audio Plug-Ins**: Find all scanned AudioUnits, VST and VST3 listed here. A matching device description will be looked for and suggested when you drop a plug-in on an **Instrument** or **Rack**. Dropping an effects plug-in on a **Rack Module** will load it as the module's insert effect.

#### 8

**DAW Drones (Audio, MIDI)**: If the DAW is open and has **Drones** loaded, these will be listed here. **Drag** one to an **Instrument** or **Rack** to claim it for your arrangement and populate it with a plug-in and device description.

 $\bigcirc$ 

**External MIDI**: Find all MIDI ports know to the system listed here. Drag one to the empty area of a **Rack**, or to an **Instrument**, if you want to send rendered MIDI output to this external hardware or software.

A port is disabled when it is already in use. You can edit the device description using the **Parameter Inspector** on the **Sound** tab to add more channels to the device.

••

**Global Rack Modules**: Find all rack modules of the current **Global Rack** here. Drag one to an **Instrument** or into a **Rack** to use a copy of it in your arrangement.

夏

**Global Instruments**: Drag one to an **Instrument** to use it in your arrangement. The advantage being that your arrangement or Sketch will work on any installation of Synfire, not depending on specific plug-ins.

**†** 

**Global Mix**: These are placeholders for a global MASTER and AUX rack module. Drag one to the empty area of your **Global Rack** to add it to the final mix.

### **Reclaiming a Drone**

When you open a DAW project with **Drones** loaded, these will connect to Synfire and appear listed in the rack module library. Those that are still available can be used by your arrangement, but some Drones may be disabled (grayed out) because they are already in use or belong to a different arrangement.

If a Drone belongs to an arrangement that you no longer have access to, or that happens to be broken for other reasons, you have the option reclaim the Drone and add it to the current arrangement. Select it in the browser and click on **Lock** icon to unlock it and add it to your arrangement.

# **I** DANGER:

You should do this only to repair a broken setup, or when the link between DAW project and arrangement somehow got lost. Once a Drone has been reclaimed, the original arrangement that owned the Drone will no longer find it.

#### **Related information**

Rack Module Preset (on page 38) Rack Module (on page 131)

# **Rack Editor**

There are three types of racks. <u>Arrangement Racks (on page 24)</u>, <u>Library Racks (on page 149)</u> or the <u>Global Rack (on page 39)</u>. Their <u>Rack Modules (on page 38)</u> (aka Modules) appear listed in this view.

Arrangement 🗮 Arrangement Rack 🖉 Device Descriptions 🕓 Repository	
All MIDI Embedded Engine1 🖉 🎭 🗮 Ċ 🚈 🖂	
© Engine1:04 ✓ Kontakt 5 AU General Purpose:A Banc ✓ ▷ Insert ▷ Insert ○ Gain ○ Gain ○ Gain ○ INF dB ⊙ ▲UX Send S M General Purpose:A Band	Rack Module
Image: Second	Device Description: General Purpose:A Band * 🖉 2 variants, 27 sounds on fixed channels
Image: Soundcase FluidR3       Soundcase AU         Soundcase FluidR3       Image: Soundcase AU         Image: Soundcase FluidR3       Image: Soundcase AU         Soundcase FluidR3       Image: Soundcase AU         Image: Soundcase FluidR3       Image: Soundcase AU         Image: Soundcase FluidR3       Image: Soundcase FluidR3	1 ✓ Classic Bass
	3     Contemp Analog - Sub I       4     E-Piano       5     EP 10-Piano       6     Jazz Guitar       7     Rock Guitar       8     Jazz Organ       9     Muted Trumpet
	10 ✓ Street Knowledge Kit
	12     Toys-Rapman Vibrapho       13     String Melody - Orchest

## **Window Toolbar**

# 

Opens the **Rack Module Library** (on page 47) on the left sidebar where you can search for **Rack Module Presets** and other items and drop them on the rack to add a new **Rack Module** (on page 38) or load a plug-in and device into an already existing module.

# **Rack Toolbar**

### All, MIDI, Embedded, Engine1

Select one of these to focus on the modules hosted on that type of port. This selection also determines which host a new module is added to when you double-click in the empty area of the rack.

# 0

When active, the arrangement remembers the **<u>Global Rack</u>** (on page 39) that is currently open. Next time you open the arrangement, you will be asked whether to also load that global rack.

# **1**

Loads the Global Rack (on page 39) that was open at the time this arrangement was last saved.

# ••

Open the current <u>Global Rack (on page 39)</u> in the <u>Audio/MIDI Setup (on page 156)</u> app.

# G

Reload all rack modules. This may help to restore a more stable state if something went wrong.

# ŲΞ

Sort all rack modules by type and port.

 $\mathbf{\nabla}$ 

Opens a pop-up menu to control the **Audio Engine** (on page 41) and purge unused modules, devices or ports.

## **Selection, Drag & Drop**

- Select a module to see its <u>contents on the right sidebar (on page 133)</u>. Select a channel on the sidebar to open a **Keyboard Widget** where you can play a sound and examine its **Playing Ranges**.
- Drag a module to another rack to copy it.
- Drop an item from the <u>Rack Module Library (on page 47)</u> to insert a new module or load a plug-in and device into an already existing module. You can drop items between the modules.
- Drop an effects plug-in or effects preset from the <u>Rack Module Library (on page 47)</u> on an existing module to load it as the insert effect.
- Move modules up or down to change their order.

#### **Related information**

Rack Module (on page 131) Sidebar: Rack Module Library (on page 47) Device Inspector (on page 133)

# **Rack Module**

Setup a Rack Module (on page 38) to provide the Sounds (on page 37) you need.



#### **Audio Port**

Moves the module to another host or port. Select an **Audio Engine Port** or **Drone** as the target where you want the module to be moved. Learn how to move a module to a Drone in a DAW <u>here (on page 239)</u>.

#### **Output Port**

Pick an external MIDI port for the module to send its output to (MIDI modules only). Any port may receive output only from a single module.

#### **Device Description**

Select a known **Device Description** that matches the plug-in or external hardware, or perform one of these actions:

### Detect

Matches the current state of the plug-in with previously saved **Rack Module Presets** and select the device description associated with those.

#### **Unknown Device**

Go without a device description for now.

#### **Create New**

Create a new device description (on page 234) for use with this module.

**Extract From Plug-in ...** 

Attempt to extract a device description (on page 236) from the plug-in.

#### Download From Repository ...

Search the **Online Repository** (on page 182) for a device description that matches the current plug-in.

#### **Show Device Layout**

Opens an inspector showing the channel layout of the device you selected for the module.

#### **Show Insert Module**

Opens an inspector for you to setup an **Effects Plug-in**. One insert effect per module can be chained *after* the plug-in's output. A device description is not needed unless you want Synfire to be able to select different programs.

### *i*) Tip:

Racks offer basic audio processing only. For production and mixing you will want to move your project to a DAW.

#### **3-Way Plug-in Button**

This button allows for three tasks to be performed:

- 1. Load or save a **Rack Module Preset** (*on page 38*). New presets are listed in the **Rack Module** Library (*on page 47*) automatically.
- 2. Open the current Plug-in Editor.
- 3. Load an Audio Plug-in.

#### Gain

Adjust the overall audio level for this module. Don't use this for mixing. Keep at 0 dB unless you need to compensate for grossly unequal audio levels.

#### **AUX Send**

Send a portion of the output to the AUX Module (on page 39), if the Global Rack has one.

### Solo

Temporarily mutes all other rack modules.

#### Mute

Temporarily mutes only this rack module.

#### **Colors, Icons**

You can customize a rack module to improve your oversight.

#### Label

Enter a name for your rack module.

# **Option Menu**

#### **Exclude From Rack**

Detaches the rack module from the current rack, so you can attach it to another. Makes it a <u>Transient</u> <u>Rack Module (on page 39)</u> that is no longer saved with the current rack.

### **Include With Rack**

Attaches a Transient Rack Module (on page 39) to the current rack, making it a permanent part of it.

#### Commands

- Drag a rack module to another rack to copy it.
- Keyboard shortcuts for Cut, Copy, Paste or Duplicate also work on a rack module.

#### **Related information**

Special Rack Modules (on page 39) Rack Module Preset (on page 38) Sidebar: Rack Module Library (on page 47) Relocating From Engine to DAW (on page 239)

# **Device Inspector**

Shows a brief overview of the **Device** used in a **Rack Module**, listing its channels and which sounds are selected and used at this time.

Select a channel to make a keyboard controller appear, allowing you to play or preview the sound.

Ra Labe	ick 1:	Module					
<b>Nor</b> Devic	ne ce D	► escription:					
Soundcase FluidR3 GM, multi-timbral, 159 programs, 6 CCs, Cognitone Soundcase (AU)							
1		Yamaha Grand Piano	Used				
2		Synth Strings 1	Used				
3	~	Acoustic Bass	Used				
4		Steel String Guitar	Used				
5		Standard	Used				
6							
7							
8							
9							
10		(Reserved Channel)					
11							
12							
13							
14							
15							
16	~						

# Ø

Jump to the device editor for more details, or in-depth editing of the **Device Description**.

# **Global Instruments Page**

On this page you configure the six <u>Global Instruments (on page 40)</u>. This configuration is saved as part of the Global Rack file.
File Edit View Playback Wind	dow Help				
< > 👘 Audio	$\bigcirc$ MIDI $ ightarrow$ Inputs	📰 Global Rack	📾 Global Instruments	${\mathscr Q}$ Global Device Descriptions	The Repository Repository
					_
Piano Soundcase (AU)	Soundcase FluidR3	Yai	maha Grand Piano	🗧 🙃	Piano
Chords Soundcase (AU)	Soundcase FluidR3	Syr	nth Strings 1	<mark>ା  ା</mark> ଜ	Global Rack
Bass Soundcase (AU)	Soundcase FluidR3	Y 🎝 Ac	oustic Bass	<mark>_</mark> ଜ	Soundcase AU
<b>Guitar</b> Soundcase (AU)	Soundcase FluidR3	🖌 🍾 🖌	eel String Guitar	<mark>_</mark> ଜ	Pan
Drums Soundcase (AU)	Soundcase FluidR3	🖌 🎸 🖌	andard	<mark>_</mark> ଜ	∞
Metronome Soundcase (AU)	Soundcase FluidR3	🖌 🔑 Sta	andard	<mark>ା  </mark>	
.= = = ⊘ ₫ © @ ⊠				Reset to Defaults	⊚
<					

# **Selection & Configuration**

Select an instrument and edit its configuration on the front panel and on the sidebar. Hit the space bar to render a preview phrase with the selected instrument.

# **Rack Module Menu**

Select the global rack module that provides the sound for the instrument.

# ß

Open the **Sound Wizard** (on page 193) to search for or create a rack module, device and sound for the instrument.

# **Status Indicator**

The square is lit to indicate the instrument's current status.

- 1. Green: Sound is online and ready.
- 2. Orange: Desired sound could not be found. A replacement is currently used.
- 3. Red: Desired sound could not be found and neither a replacement.

# $\mathbf{n}$

Preview the instrument with a phrase suitable for its Category (on page 34).

## Volume, Pan

These are defaults when the instrument is used for global tasks. If an arrangement references a global instrument, the arrangement may have its own <u>Volume (on page 318)</u> or <u>Pan (on page 292)</u> parameters in place. Click on the small X to clear the assignment.

# **Playing Ranges**

# **Reset to Defaults**

Open this pop-up menu and select a preferred device to provide default sounds for all instruments. This works best with GM compatible devices or devices that have categories assigned to all sounds.

#### Toolbar

Select and adjust Playing Ranges (on page 57) as explained here.

# **Devices Page**

Under the tab **Device Descriptions** you manage your collection of **Devices** (on page 36). Devices managed inside an **<u>Arrangement</u>** (on page 19)</del> are a part of that arrangement. They can't be used by other arrangements unless you copy them with **Device > Save to Global Rack**.

Devices managed by **Audio/MIDI Setup** are global. They can be used by all **Global Racks** and copied into **Arrangements**, **Sketches** or **Libraries**.

< > ◧▯‰▨И▢॥▷೫◶◯깥 01:00:00.00 ۪ቋ₄◙ ∞ 之ଓ▣▦▦▥▯;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;	
Structure Overview Matrix Progression Harmonizer Palette Snippets Library Sounds	
The Pale Rider (Snippets)	
Device Descriptions         Properties         Variants, Channels         Programs           Energin         © C         Device Name         Variants	
Image: Control Ubraphone Vbraphone	orted

# **Saving Devices**

Synfire automatically saves pending changes of all devices to disk. With the **Audio/MIDI Setup** window, you can also save a device manually with *File > Save As ...* to a different place.

# **Device Browser**

This browser helps you organize all your **Device Descriptions** in folders. Global devices are maintained by the **Global Rack**. The folder structure is mirrored in the file system, so every device is an actual file on disk.

The devices that were copied into an **Arrangement Rack** however are stored internally with the arrangement. You won't find them on your hard disk as separate files.

## **Related information**

Devices (on page 36)

# **Device Properties**

On the tab Properties you can edit general settings of a device.



#### **Device Name**

The name of the device is used to display it on the list and elsewhere in the software. You can change this name at any time.

# Manufacturer

This designation should contain a short name for the manufacturer, e.g. **NI** for Native Instruments, or **YAMAHA**. Please always use exactly the same name for the same manufacturer.

#### Model (Plug-in)

Name of the hardware model or plug-in: JV2080, MU90R, EWQLPlay, Kontakt6, etc.

#### Version (Instance)

In case it's a plug-in, put a designation for the loaded preset here that is described by the device description: My Orchestra Kit, Or Misc Basses. You can leave this empty for hardware synthesizers.

#### Сору

This number is incremented every time a device is copied or cloned. It is for your information only, so you can determine which of multiple copies of a device is probably the most current.

# CAUTION:

Manufacturer, Plug-in/Model and Version are used to create a unique file name the device is saved to. Where possible, you should not change this identifier anymore after you began using it in your projects. In order to change the identifier, you need to unlock the input fields first. Synfire will propagate the change to all currently open files to ensure that instruments still point to the same device.

#### Latency

Enter a value in milliseconds by which MIDI data shall be sent ahead of time in order to compensate for the device's latency.

## Template

Check this box, if this device description is supposed to be a **Device Template** (on page 37).

## Effect

Check this box if the plug-in or hardware is an effects processor.

#### Generic

Indicates this device was created automatically when one was needed, in order to serve as a starting point. You may remove this box and edit the device to fully suit your needs.

#### Send Default Volume, Pan, Reverb on Sound Selection

Whether to automatically send <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u>, <u>Reverb</u> and other mixingrelated CC whenever a sound is selected. Disable this if Synfire shall not interfere with a mix you manually set up in your DAW or with external hardware.

## **Reset MIDI Controllers On Sound Selection**

Some devices require that MIDI CC be reset to default values after a new program/patch was selected. If you notice that your device is behaving strangely with respect to sustain pedals, pitch bend or modulation, you may need to check this option.

## **Re-Send Repeated Articulations**

Send <u>Articulations (on page 201)</u> every time, regardless whether the previously sent articulation has not changed.

## **Allow Modification In Arrangement**

Whether you want the **Instrument Inspector** of an arrangement to show the full details of the sound you selected for an instrument. Although this spares you a visit to the device editor on the **Sounds** page, it adds a lot of information that might distract you. This option is always disabled for global (shared) devices.

## **Custom Controllers, Articulations**

Edit the <u>Custom Controllers (on page 199)</u> or <u>Custom Articulations (on page 201)</u> available for this device.

# **Updating an Embedded Device**

If a device was originally copied into the arrangement, you can update it to the current version of the <u>Global Device</u> <u>Description (on page 39)</u>. This imports a new copy into the arrangement.

# **Device Variants & Channels**

The MIDI ports and channels of a **Device** are configured on the **Variants, Channels** tab. One device may support more than one port and channel layout, called **Variants**.

			Properties	Varia	ants,	Channels	Programs						
EW	QL	Orch Lean:A Strings		~									
De		e not currently used b	oy global rack							Fixed S	Sound		
1	~	CBS KS Master				Contrahass				Sound Nam			
-		VCS KS Master		ě		Collo				18V KS	Master		
<u> </u>	• •			Š						Category			
3	Ň	VAS KS Master		$\sim$		viola				Violin			~
4	~	11V KS Master		8		Violin		✓ ⊗		Method of S	Selection	MSB	LSB
5	~	18V KS Master				Violin		< ⊗		Program	Only 🗸	0	0
6	~	SVC KS Master			$\geq$	Cello		<ul><li>✓ ⊗</li></ul>		None	~	GM	
7	~	SCB KS Master			$\mathbf{\nabla}$	Contrabass		<ul><li>✓ ⊗</li></ul>		None			
8	~	SVA KS Master				Viola		✓ ⊗		Con	trollers		
9	~	SVL KS Master				Violin		✓ ⊗		Artic	ulations		
$\pm$													Volume
										90	50		
										A single so assigned to	ound is perman o this channel.		T
			$\odot$										
<									>				

For most devices a single port with 16 MIDI channels is sufficient. For comprehensive sound libraries, you may want to add up to four **Variants** A, B, C, D. Each variant represents a different configuration, operation mode, or preset of the same device and supports up to 16 MIDI channels. The advantage of using variants versus creating multiple devices is that <u>Custom Controllers (on page 199)</u>, <u>Articulations (on page 201)</u> and <u>device properties (on page 135)</u> need only be defined once for all variants.

Every MIDI channel can be set to one of three types:

#### **Fixed Sound**

A single sound is permanently addressed on this channel. Its properties can be set using the inspector on the sidebar.

#### **Dynamic Program Selection**

An arbitrary program or patch of the device is selected by sending a MIDI bank and program selection message. The list of available sounds is managed on the tab <u>Programs (on page 138)</u>.

### **Reserved Channel**

On this channel only those sounds can be selected that have the same channel number set as their **reserved channel** (for example, this is the case for GM drum kits that require channel 10). These sounds are also managed on the tab **Programs** (on page 138).

Multiple channel types can be mixed in the same variant. If a device has multiple variants with dynamic channels, Synfire considers all of them for sound assignments equally.

A green indicator labeled Used indicates whether a channel is currently in use by one or more instruments, which are listed on the sidebar for your information.

# *i* Tip:

You can use this page to edit a sound's properties, controllers and articulations while the plug-in is loaded and then call *Copy Sound To User Bank* from the options menu to save it permanently to the **Programs** page. This is handy when you have many sounds to add for a large library.

# i Tip:

For General MIDI (GM) synths and sound generators, we recommend you always exclude channel 10 from dynamic allocation. The standard requires it be reserved for drum kits.

## Note:

Dynamic channels are managed by Synfire automatically. When multiple arrangements are open at the same time, the current foreground window may temporarily put the sounds of a background window aside in order to ensure all sounds of the foreground window are ready to play.

# **Related information**

Dynamic Sound Allocation (on page 42)

# **Device Programs**

Programs or patches that are selected by sending a MIDI message are managed on the tab **Programs**. Sounds are organized in Banks and classified by **Category** (on page 34).



# **Sound Banks**

The MIDI standard allows a maximum of 128 sounds per bank. Sound banks are most common for hardware synthesizers and their software emulations. If your device doesn't support sound banks, create a single bank that takes all sounds.

Use the + and - buttons of the banks list to add or remove banks.

## Name

Be sure to name each bank.

# **Method of Bank Selection**

For audio plug-ins, the method **Plug-in Preset Selection** usually does the trick. Please consult the MIDI documentation of your device about the MIDI messages sent to select a particular bank and patch/program. Pick that type of message from the drop-down menu. The variables **MSB** and **LSB** are placeholders filled in by the particular sound.

# **GM (General MIDI)**

Check this box if all sounds in the bank conform to the **General MIDI Standard** (GM). The standard defines a set of program change numbers, names and categories, sparing you the effort of setting up

the sounds yourself. Use the list's options menu (right-click) *Initialize General MIDI* to populate an entire bank with sounds according to the standard. Already existing programs will be preserved.

# Categories

This tree shows a hierarchy of Instrument Categories (on page 34). They are basically like folders.

Drop one or more sounds on a category to assign that category to them.

With *Option > Preview Phrase* on the list, you can cut, copy or paste any <u>Phrase (on page 17)</u> you want to use to preview sounds of that category.

# Sounds

The list of sounds can be sorted by **Name**, **Program**, **Category**. You may select one or more sounds to edit their properties on the sidebar inspector.

- Select one or more sounds to edit them as a group together.
- Drag one or more sounds to a different **Sound Bank** to move them. Hold down 🔨 Ctrl to copy them.
- Use keyboard commands Cut, Copy, Paste to copy or move one or more sounds to and from other **Devices** or **Sound Banks**.

# **Creating New Sounds**

Do the following to add a new sound to the list from scratch.

- 1. Select the bank it should be added to.
- 2. Select an appropriate category in the tree.
- 3. Click on + in the list toolbar.

Now fill in the form to configure the sound.

# **Sound Properties**

Edit the properties (meta-data) of a <u>Sound (on page 37)</u> on the <u>Parameter Inspector (on page 54)</u> or the Sounds page.



 $\otimes$ 

Browse **<u>Template Devices** (on page 37) for already known sound properties in order to copy them into this sound.</u>

## Sound Name

The name that shows up next to your **Instrument** (on page 33) and when you browse for sounds.

## Category

Pick a <u>Category (on page 34)</u> from the menu, or drop one or more sounds on the desired category found on the tree.

#### **Method of Selection**

How the program/patch represented by this sound is <u>selected (on page 138)</u> (optional).

#### **Preferred Channel**

Set only if the sound requires or prefers a specific channel. If this is a reserved channel, also tick the box **Reserved**. If the sound is always found on the same channel, you must edit the channel on the **Variants, Channels** (on page 137) tab, not here!

#### Program

The number used to select the sound with a MIDI program change message (1 - 128). Check the GM box, if program change and instrument category conform to the **General MIDI** standard.

# **Controllers, Articulations**

Setup optional <u>**Custom Controllers** (on page 199)</u> and <u>**Articulations** (on page 201)</u> for the sound. Those defined for the device already need not be redefined.

# Velocity, Volume

These are optional preferences used only for previewing phrases when no such values are available otherwise.

## Latency

Sound latency (milliseconds). Added to the device latency.

# **Additional Information**

Explains why a sound is not currently audible. Be aware that if you edit a device description, it is not necessarily "online", i.e. currently used by a **Rack Module**. If so, you can't hear the sounds you are editing, of course.

# **Sound Details In Arrangement**

Some properties of a sound can be made visible and edited on the **Parameter Inspector**. This form is usually readonly (grayed out) unless you enabled **Allow For Modification in the Arrange View** with the device description. Device descriptions created by the **Sound Wizard** for convenience do allow this by default, so you can easily alter a sound's properties without visiting the device editor.



# C

Jump to the device editor where you can edit the sound and all other sounds of the device in context.

## **Related information**

Instrument Properties (on page 55)

# Repository

Cognitone provides a collection of **Device Descriptions** that you can download from inside Synfire. On the tab

Search Results	✓ Q Search			$\otimes$			
Garritan: GPO5	Garritan PO 5 Template	РТ	****	Alpho	542 (80% edited)		
Garritan: JABB	Garritan Jazz and Big Band		***		68 (85% edited)		
GreenOakSoftware: Crystal	Crystal	Р	***	timeline	114 (79% edited)		
IKMultimedia: MODOBASS	MODO BASS	Р	**	Gastod	1 (20% edited)		
Internal: ExternalGlobals	External Global Instruments		****	timeline	6 (80% edited)		
Internal: ExternalGlobals	External Global Instruments			kiredg109	6 (100% edited)		
Internal: ExternalGlobals	External Global Instruments (GM)	GM		timeline	129 (80% edited)		
Internal: ExternalGlobals	External Global Instruments (GM)	GM	***	horn	128 (100% edited)		
Internal: SoftwareGMSynth	Internal GM Synth	GM	****	timeline	129 (80% edited)		
Internal: SoftwareGMSynth	Internal GM Synth	GM		juancfpuerta	127 (80% edited)		
Internal: SoftwareGMSynth	Internal GM Synth	GM	***	cledut	129 (80% edited)		
KAWAI: K1	Kawai K1	м		timeline	64 (20% edited)		
KAWAI: K11	Kawai K11	м		timeline	135 (20% edited)		
Korg: Karma	Korg KARMA	GM		timeline	905 (29% edited)		
Korg: Kronos	Korg Kronos GM	GM	****	timeline	128 (100% edited)		
Korg: KronosV3	Korg Kronos v3	м		bmanpro	5757 (50% edited)		
Korg: LegacyCell	Korg Legacy Cell (VST)	Р		bmanpro	128 (80% edited)		
Korg: M1	Korg M1	Р	***	timeline	1533 (20% edited)		
Korg: M164	M1	Р	****	bmanpro	50 (80% edited)		
Korg: M164	M1-64	Р	****	bmanpro	1633 (80% edited)		
Korg: MonoPoly	Korg MonoPoly	Р	****	timeline	256 (100% edited)		
Korg: MS20	Korg MS-20	P -	****	timeline	32 (20% edited)		
Korg Korg MonoPoly, Preset: "Fact	torvPresets"				29.07.2017	100%	
Unloader Comments:	,	Summary					Please Vote:
KORG MonoPoly factory presets.		256 programs, 1 Filename: Korg_1 KB)		lited, KORG Mono toryPresets.devic			Excellent Very Useful Ok Insufficient Useless Or Broken
							Download Device

Repository, you have access to this database.

The database allows all users to upload their devices and share them. Ask others if they possibly have a device description you need and whether they would upload it. Upload descriptions yourself that you think could be useful for others. Doing so will grow the collection quickly to benefit all users.

# Search And Download

You can browse the repository by different criteria. Use the drop-down menus on top of the list for this. Enter the name of a plug-in, manufacturer or model into the search field to look for matching descriptions. After downloading a description, it is saved to your list of global device descriptions.

This is how you quickly find a description for a particular plug-in:

- 1. On the **Rack** tab, select the rack module that is loaded with the plug-in you need a description for.
- 2. Now switch to the **Repository** tab. A search for the plug-in is run and potentially suitable descriptions will show up on the list.

# Uploading

First check if the description is labeled correctly and populated with sounds to an extent that makes it useful. It need not be finished, but should at least include enough information for others to eventually complete it. If necessary, Synfire will show you a warning message with potential sources of errors before you upload.

Use the comments fields of the description to tell others which data is already included and what's still missing. Others may want to complete your description and upload an updated version.

If you are uploading a description for a plug-in, you should associate it with a plug-in instance with *Options > Associate With Plug-in...*. This is not necessary, if the description was originally extracted from a plug-in. Without this association, others users will have problems finding the description. Manufacturer and Model are critical for a successful search.

Finally, with Options > Upload to Repository you can submit the description to the database.

If you upload the same description again later on, it will replace your previous upload. Other users will see an indicator showing them there is a newer version available for a description they already downloaded.

# i Tip:

If you see descriptions that are seemingly incomplete, do not hesitate to download and edit them. If you have a better or more complete description, upload it. You will never overwrite an already existing description in the database. Multiple versions of the same device can coexist.

# Panels

# Standalone widgets floating on your desktop

Some views included with Synfire can also be opened as separate panels tacked to the desktop, so you can observe them while you play and work.

# Circle Of Fifth

This widget shows all traditional keys arranged in a circle. The major keys are on the outside, and their parallel minor keys are on the inside. Their root notes are a 5th apart, hence the name.

You find this widget on the sidebar of the **Progression** or **Palette** tabs. Alternatively, open a standalone widget with *Window > Circle Of Fifth*.



The letters **T**, **D**, **s** denote the functions **Tonic**, **Dominant**, **Sub-dominant** respectively. They indicate which function the currently playing chord has in that key. This way you can spot at a glance in which keys the current chord has a primary function. This is relevant when you want to change keys.

- Click on a key to navigate to it.
- Click on a key while holding down  $\sim$  Alt to keep the current (alternative) scales.
- Double-click on a key to open a Palette on it.
- Use the <> buttons to navigate back and forth along recently visited keys.
- Hold down **1** Shift while you select a key in order to highlight all chords it has in common with the previous key. This is helpful for a key change (on page 10).
- Hold down **#** Control while you click in order to stay in the current key and highlight all chords it has in common with the key you clicked on. Moreover, this highlights the chords of that key in all currently open **Palette** windows.

# 7) Tip:

All temporary highlights can be locked in place with the **Freeze** switch, so you can play the palette without resetting the current display.

# 🚺 Tip:

If you have difficulties reading multiple chord highlights, use the black and white coloring preset for **Printing**.

# Note:

In traditional major/minor tonality, these are all the relevant keys you should care about. Although there exist more remote keys with additional accidentals, those don't matter for **12 Tone Equal Temperament** tuning. You can't play them on a regularly tuned piano or MIDI keyboard anyway, without re-tuning all your equipment, which is a mess.

## **Related information**

Keys (on page 9) Harmonic Functions (on page 319)

# **Keyboard Widget**



This keyboard highlights the currently playing **Chord**, bass and **Vertical Scale**. Open additional panels with *Window* > *Keyboard* at any size.

# Scales

Show Vertical Scale instead of Chord.

#### **All Octaves**

Extend scale highlighting to all octaves.

## **MIDI Input**

Display live MIDI input. If you want a palette to respond to keys presses here, also check **MIDI Input** with the palette.

#### **Mouse Wheel**

Scroll octave range.

Set or unset keyboard split for live input.



For playing along on a piano or synthesizer, you can change the size to make fingerings easily readable from a distance.

# **Related information**

Setting Up MIDI Input (on page 233)

# Scale Board

With this special keyboard, you can add segments to a Figure (on page 294) incrementally. Select the desired position in the Figure (on page 294) and click the keys to add symbols. It takes some getting used to.

# String Instrument Panel

Open as many instances you want and configure them in different ways. To play along on an instrument, you can change the size to make the fingerings easily readable from a distance.



## Tuning

Pick a preset tuning for your instrument.

# + -

Add or remove additional frets.

# Mode

Switch between Chord and Vertical Scale.

< >

Browse a variety of fingerings that exists for the chord.

# 7 Tip:

Open several panels and use them to play along with other musicians.

# Legend

## **Open String**

A string without a green dot is meant to be played open.

# Χ

String should be muted. It is not meant to be played as an open string.

## **Blue Square**

This string plays the **Root** note of the chord.

#### Barré

Green bar with a number next to it which indicates the fret position.

# Note:

Some chords have more notes than there are strings. Synfire attempts to preserve especially characteristic intervals of a chord.

# **Preferences**

Synfire remembers your last selection for each chord. You can also save and load these manually with *Preferences > Defaults > Personal Fretboard Preferences*.

# **Supported Instruments**

Instrument	Tuning
Guitar	EADGBE
Left-handed Guitar (lefty guitar)	EBGDAE
Bass Guitar	E A D G
Ukulele	GCEA
Viola da Gamba	D G C E A D
Cello, Banjo, Tenor Guitar, Viola	C G D A
Cello, Violin, Mandolin, Irish Banjo	G D A E
Tenor, Bass or G Violin (Lute)	G C F A D G

# Note:

You can add your own tunings to the configuration folder **Tunings** (simple XML files).

# Harmonic Context Panel

Detailed information on the <u>Harmonic Context (on page 12)</u> that is currently playing. In addition to the properties of the harmonic context, this panel also lists the chord extensions that can be played without disrupting harmony.

$\stackrel{\texttt{left}}{=} \bigcirc \oslash$	C9 > Gm	(add9)	
Automatic Bass	~	Automatic Inversion	~
Chord	Gm(add	9)	
Vertical	G.doria	า	
Horizontal	F.major		
Bass	-		
Кеу	C.Major		
Relation	F.Major-	minor	
Numeral	v		
Function	d		
Extensions	m6(add9)	m9 m11 m13	

Drag this panel to any **Progression** (on page 14) to insert the chord and all its context.

Open a floating panel with Window > Harmonic Context.

## **Related information**

Harmonic Context (on page 12) Progressions (on page 14)

# Dialogs

# Dialog windows that pop up to accomplish a certain task

The topics under this section are all entry points for the embedded **Help Browser** where they show up when you hover over or select objects in the application. We therefore suggest you open Synfire and do just that, instead of reading these sections like chapters in a user manual.

# Preferences

# License

This page contains information on your license. The information you provide on copyright, editors and claims is added securely to the documents you save.

# **User Interface**

Some of these settings take effect only after re-opening a window.

# Language

Select your language or the language you are able to sufficiently understand. This preference is read form the operating system at installation, but can be altered at any time.

## Colors

Choose among several color themes.

# **Tool Tips**

Brief pop-up help that shows up when you move the mouse over a control or area of the user interface. For more comprehensive help, open the **Embedded Help** browser.

## **Local Menu Bars**

On macOS the main menu bar appears at the top of the screen. Enable this if you want it to appear at the top of each window, which is common on Microsoft Windows and has some advantages when you have many windows open at the same time.

## **Native File Dialogs**

If you experience issues with file dialogs provided by the operating system, you can disable this to use custom dialogs provided by Synfire.

## **Disable Widgets in Background**

On macOS it is common to gray out widgets when a window isn't in the foreground. Disable this if you can do without this visual feedback and get a faster response.

# **Reset Dismissed Warnings**

Show all warnings again, even if you have dismissed them in the past.

# **Reset Playhead on Container Selection**

Move the playhead to the beginning of a container when it is selected.

## **Reset Playhead on Parameter Data Selection**

Move the playhead to the last selection established in a <u>Parameter View (on page 144)</u> (with a little pre-roll).

## **Select New Container After Creation**

Immediately select a newly created container.

## **Don't merge Adjacent Equal Chords**

Allow repeated instances of the same Harmonic Context to appear in a progression.

# Edit Keyboard Shortcuts

Open a dialog for editing keyboard commands (on page 210).

# **Files**

The installation and configuration folders can't be changed retroactively.

# Configuration

Synfire saves your preferences, setup, templates and racks in this folder. The original settings are kept in the installation folder where they are protected from being accidentally overwritten.

## **Projects**

This where your racks, settings, arrangements, progressions, palettes and other documents go by default.

## **Number Of Backups**

The number of recent saves to keep, in case you need to revert to an earlier version.

## **Automatic Backup**

Save a backup of the currently open arrangement every couple minutes that you can recover from in case of a crash.

## **LilyPond Engraver**

Installation path and command line options for invoking LilyPond notation export.

## **Notation**

Configure three command lines for processing exported notation files.

- 1. **Convert**: This command runs a program on the file exported by Synfire. This may be a command line utility that renders a PDF, JPG or other format from the exported file.
- View: This starts a program to display the output generated by the conversion program or the exported file directly. For example, this may be a PDF viewer or a notation software that can open the exported file directly. This command is only run when you enable it.
- 3. **Debug**: This starts a program that can be used to examine the exported file, for example with an XML editor. This command is only run when you enable it.

The following variables are defined as placeholders for the command line templates.

## {filename}

The absolute file name of the exported file with directory and file extension.

## {name}

The name of the file without the file extension and directory path, e.g. Lullaby.

## {ext}

The file extension of the file, e.g. ly or .xml including the dot.

# {folder}

The directory of the exported file, without a trailing slash.

#### {options}

A series of command line options specific to the format, e.g. --dvi --backend=tex. These are provided automatically depending on your output format selection. If you need different options, you can add them to the command line template directly (LilyPond only).

#### {format}

The file extension of the output format, e.g. .pdf, including the dot (LilyPond only).

#### {output}

Full absolute file name of the file created by the conversion program, i.e. the PDF, JPG or whatever is being compiled (LilyPond only).

## {log}

Full absolute file name of the log file created by the conversion program (LilyPond only).

# CAUTION:

The **Convert** command blocks Synfire until it has completed. If you open an application with a user interface rather than a command line utility, you should do that with the **View** command.

# **Defaults**

#### Constants

Several values to use when they are not available from context. They are mostly used to render previews for chords, scales and other basic objects.

#### **Preview (Outlet)**

A global preview progression used to render phrases that don't provide their own <u>Harmony (on page</u> <u>300</u>).

#### **Tolerate Triple Sharps/Flats**

Allow three accidentals to occur with notation export and with internal calculations. Default is disabled.

## Use German H for B and B for Bb

Enable only if you need to comply to German notation traditions. Read more on this here (on page 325).

# **Personal Fretboard Preferences**

Synfire remembers the last used fingering patterns for every chord you selected on the **String Instruments** panels for guitar, bass, violin, etc.

# **Scale Selection**

Global Scale Selection Preferences (on page 192) that are copied into every new document by default.

**Related information** 

Scale Selection Preferences (on page 192)

# **Scale Selection Preferences**

Synfire automatically selects suitable scales for a <u>Harmonic Context (on page 12)</u> if only a chord is known so far, or when a new estimation of keys and scales is requested. This selection is guided by the built-in knowledge base and your **Scale Selection Preferences**.

The global defaults for these settings can be found at *Edit > Preferences ... > Scale Selection*. These are copied into new documents (Arrangements, Palettes, Progressions) where you can edit them on the **Scale Selection** tab on the **Parameter Inspector** for **Harmony**.

Context	Scheme		000					
Scale Selection Preferences								
Traditional								
Alternative	e							
Strict								
<ul> <li>Prefer St</li> <li>Include F</li> <li>Include A</li> <li>Create a</li> <li>Smooth</li> </ul>	<ul> <li>Prefer Standard Modes</li> <li>Include Horizontal Scale Variants</li> <li>Include Alternative Scale Sets</li> <li>Create ad-hoc Horizontal Scales</li> <li>Smooth Transitions</li> </ul>							
Estima	te							

# **Prefer Standard Modes**

Favor traditional major and minor <u>modes (on page 9)</u> over other scales. Facilitates a more customary sound even with <u>alternative palettes (on page 27)</u>. If you want to exploit the full tonal richness of an alternative palette, you should turn this off.

### **Include Horizontal Scale Variants**

Allow Synfire to extend a <u>Scale Set (on page 11)</u> with known variants of a scale dynamically as needed. For example, <u>hungarian-minor</u> is considered a member of a standard minor palette in this operating mode.

## **Include Alternative Scale Sets**

Consider dynamically generated <u>Scale Sets (on page 11)</u> as alternatives for traditional Scale Sets (keys). This augments the number of available scales.

#### **Create ad-hoc Horizontal Scales**

Enable Synfire to dynamically generate new <u>horizontal scales (on page 8)</u> as needed. Disable only if you want to stick to traditional harmony rather strictly.

#### **Smooth Transitions**

Minimize pitch leaps when moving from one scale to the next. Scales with many tones in common are then favored. When disabled, you will get more contrasting changes.

#### **Related information**

Harmony Concept (on page 6)

# Sound Assignment

There are multiple ways of assigning a sound to an instrument.

# **Related information**

Assigning Sounds (on page 232)

# Sound Wizard

This dialog guides you through the steps of selecting a **Sound** for an **Instrument**. It assists you with creating new <u>**Rack**</u> <u>**Modules** (*on page 38*)</u> and <u>**Device Descriptions** (*on page 36*)</u> if needed. Depending on context, a hint may show up under the title, suggesting a specific sound to select by default.

# **Home Page**

Sound	d Selection Wizard Step 1 / 3				?		
Suggestion: Rack Module Preset <b>Kontakt NI Pianos</b> provides sound <b>NI Noire Felt</b> and device <b>Kontakt NI Pianos</b>							
Yamaha Grand Piano							
Browse Available Sounds							
New Rack Module							
Audio Plug-In	Engine1	~	Kontakt NI Pianos	~			
C Drone in DAW (Audio)		~					
C Drone in DAW (MIDI)							
External MIDI Instrument	Embedded GM Synth	~					
Copy Global Rack	Soundcase FluidR3						
Embedded GM Synth							
New Channel On Rack Module							
Global Instrument		~					
Cancel			1	lext			

#### **Browse Available Sounds**

Opens the <u>Sound Browser (on page 197)</u> page where you can select any of the sounds provided by device descriptions on your racks. If you want to add a new sound that is not yet listed, select one of the options below.

#### **New Rack Module**

Adds a new **<u>Rack Module** (on page 38)</u> to your rack.

#### **Audio Plug-In**

Load a VST, VST3 or AudioUnit plug-in into the Audio Engine. Select a plug-in or <u>Rack</u> <u>Module Preset (on page 38)</u> to load before you proceed.

#### Drone in DAW (Audio)

Load a VST or AudioUnit plug-in into a **Drone** (on page 43) inside your DAW. Select a plugin or **Rack Module Preset** (on page 38) to load before you proceed. This is only enabled if Synfire finds at least one unused (reserved) Audio Drone in your DAW.

## Drone in DAW (MIDI)

Use a <u>MIDI Drone (on page 44)</u> to feed rendered music into your DAW with accurate timing. This is only enabled if Synfire finds at least one unused (reserved) MIDI Drone in your DAW.

# **Copy Global Rack**

Copy a global rack module into your arrangement where you may customize it as you need.

## **External MIDI Instrument**

Select a MIDI port if you want to send the rendered music to an external hardware or software.

#### **Embedded GM Synth**

Send rendered music to the <u>built-in synthesizer (on page 158</u>). It needs to be enabled in **Audio & MIDI Setup**. Only a single module with the Embedded GM Synth can exist on any rack. Once such a module was added, all of its sounds are available to browse and select for up to 16 different instruments.

#### **New Channel On Existing Module**

Configure a new MIDI channel for an already existing rack module. Enabled only if a module uses a device that provides fixed channels.

#### **Global Instrument**

Use a reference to a **Global Instrument**. Recommended for Sketches, so they will work with any user's setup.

Proceed to the next page with the Next button.

# **Device Description**

Configure the meta-data required for Synfire to render music that works with your sound. Open the plug-in editor with the **Plug-in Button**, if you want to compare your settings against the actual state and contents of the plug-in.

	Device	e Description		$\sim$				
	Step 2 / 3							
	Ма	noPoly VST						
Sugge descri	Suggestion: Open the plug-in editor and make sure its contents match the device description and provide the desired sound.							
New								
<ul> <li>Fixed Cl</li> </ul>	hannels (Sampler)							
1-Chanr	nel Multiple Programs (Synth)							
🗌 16-Char	nnel Multiple Programs (Multi-Tin	nbral)						
	Device Name							
	MonoPoly Device							
Detected	Korg MonoPoly	~						
Сору	Absynth 5 Pads							
Available								
Cancel			Back	Next				

# Detected

If the content of a plug-in hints at one or more already known **<u>Rack Module Presets</u>** (on page 38), their devices will appear on this menu. Select one that matches your plug-in before you proceed.

## **Available**

Devices already present in your arrangement can be reused for new modules.

Сору

If the plug-in or external instrument matches one of your <u>Global Device Descriptions (on page 134)</u>, you can copy it into your Arrangement and customize it there, if needed.

## New

If there is no ready-to-use **Device Description** (on page 36) for the plug-in or external instrument you want to use, just create one on the fly with this option. Enter a name for that device (e.g. the plug-in name) and select its type (on page 234) before you proceed.

Proceed to the next page with the Next button.

# **Sound Properties**

On the last page you configure the properties of the sound you are about to use. Open the plug-in editor with the Plugin Button, if you want to check your settings against the actual state of the plug-in.



## Sound Name

Enter a name you will best recognize on lists and menus going forward.

# Channel

Select the MIDI channel your sound is listening on. If the device uses dynamically allocated channels, this menu is disabled.

#### Category

Select an instrument Category, so Synfire can render music that works best for the sound. If none of the categories match, select one that comes close in terms of its role in the arrangement.

#### Program

Set a program/patch number that selects the desired sound on the device. This is hidden if there are no such sounds on the device.

# $\otimes$

Browse all <u>**Template Devices** (on page 37)</u> for sound properties to copy and insert here. With large sound libraries, this saves you a lot of manual work.

#### **Playing Ranges**

Use this keyboard controller to adjust the <u>Playing Ranges (on page 35)</u> for the instrument you are setting up. If the device was created specifically for your arrangement, the ranges are also saved with the sound for subsequent reuse.

# Ø

Select this if you want the instrument to use the playing ranges of this sound once you press OK

# **Transport Bar**

Use the transport bar to test the current sound in the context of a full or solo playback, or preview it with a generic phrase that suits its category.

Finally accept the sound for the instrument with the OK button.

# **Browsing Available Sounds**

This page of the <u>Sound Wizard (on page 193)</u> lists all sounds currently provided by either the <u>Global Rack (on page</u> <u>39)</u> or the local <u>Arrangement Rack (on page 24)</u>. Using the tabs All, Global and Arrangement Rack you can filter the list's contents selectively.

The browser can only list sounds that are provided by the **Device Descriptions** you have selected for each **Rack Module**. It is not possible at this point to look into a plug-in and provide a list of all sounds it potentially has to offer. This page is therefore most useful for selecting sounds from devices that are already fully covered by a device description.

If you want to setup a new sound on the fly, please use the guided procedure of the <u>Sound Wizard (on page 193)</u> starting from its home page.

# Synfire User Manual | 2 - User Interface | 198



#### **Devices**

List of devices to browse for sounds, filtered according to the tab selected on top.

## **Banks**

Narrow down your search to a particular sound bank of the currently selected device (optional).

## **Categories**

Select to narrow down your search (optional).

## **Sounds**

Shows your current search result. Select a sound to preview it and assign it to the instrument.

# OK

Accept the selected sound.

# Q

If you know at least some of the name of the sound your are looking for, just type it into the search box.

ß

Switch to the home page of the **Sound Wizard** (on page 193) in order to interactively create or extend a rack module that provides a new sound.

# **Transport Bar**

Use the transport bar to test the current sound in the context of a full or solo playback, or preview it with a generic phrase that suits its category.

# **Browsing Sound Templates**

This browser pops up so you can lookup the properties of a sound provided by <u>**Template Devices** (on page 37)</u>. Select a template from the list in order to copy its properties to the sound your are currently editing.

The properties stored in a template include <u>Category (on page 34)</u>, <u>Playing Ranges (on page 35)</u>, <u>Custom Controllers</u> (on page 199), <u>Articulations (on page 201)</u> and program change selection methods and patch numbers.



This browser works exactly like the other <u>Sound Browser (on page 197</u>), except you can't preview the sound templates.

 $\otimes$ 

Everywhere you find this **Templates** icon next to a sound, you can call up the browser to insert sound properties copied from a **Template Devices** (on page 37).

# **Related information**

Device Templates (on page 37)

# **Custom Controllers**

Synfire allows for up to eight user-defined MIDI controllers (CC) to be defined. Custom Controllers show up in the user interface as **Parameters** (on page 2).

@ Device	Description					
	VSL Prime Edition: Stri	ings				
	Name	Controller Number		Default Valu	e (optional)	
Controller A	Timbre	8 Balance	~	127	🗸 Use	Test
Controller B	Dynamic Range	30 -	~	127	🗸 Use	Test
Controller C	Attack	22 -	~	0	🗸 Use	Test
Controller D	Release	23 -	~	64	🗸 Use	Test
Controller E	Filter	24 -	~	127	🗸 Use	Test
Controller F	Hum. Delay	25 -	~	0	🗸 Use	Test
Controller G	Hum. Tuning	26 -	~	0	🗸 Use	Test
Controller H	Hum. Amount	27 -	~	127	🗸 Use	Test
	Volume			100	🗸 Use	Test
	Pan			64	Use	Test
	Reverb			0	Use	Test
	Modulation			100	🗸 Use	Test
	Expression			0	Use	Test
	Breath			0	Use	Test
ОК	Сору			Clear		Cancel

# Name

Label of the controller as it shows up in **Parameter Outlets**. Leave this blank to inherit the parent controller definition (see: Inheritance) or disable the controller.

## **Controller Number**

Pick any of the 127 MIDI CC numbers.

#### **Default Value**

The associated parameter will use this value by default if you don't set it in an arrangement. The value is also sent when an instrument is selected (if the device has this option enabled).

# Use

Enables the default value to be used. Disable this if you don't want Synfire to send zero (or whatever value is currently entered here).

#### Test

Transmit a series of controller values to the currently edited instrument. Watch the instrument for incoming MIDI data to verify your settings.

#### Enable

Some standard controllers can also get a default value assigned. For this they need to be enabled.

## **Copy To Sound**

See section below.

## Inheritance

Custom Controllers can be defined per <u>Device (on page 36)</u>, per <u>Sound (on page 37)</u> and per <u>Instrument (on page 33)</u>. And they are inherited in that order: A controller defined for a **Device** is automatically available to all its **Sounds**, which in turn is available to all **Instruments** that are using the sound. An individual sound or instrument may add to or override the inherited definitions.

This inheritance is accessible through tabs on the configuration dialog.

# **Copy To Sound**

When this switch is enabled while you are editing the settings of an **Instrument**, the settings will also be written through to the **Sound** where they are saved permanently along with the Device Description. This is useful if you want to preserve your incremental edits for later reuse by saving a **Rack Module Preset** later on.



This feature is available with the Express and Pro editions.

# **Custom Articulations**

Synfire allows for any number of user-defined sound articulations to be defined. Articulations are used to switch between different variants of a sound (e.g. Legato, Pizzicato, Staccato). They are specific to a particular sound library, synthesizer or plug-in.

Articulations are assigned to individual Figure (on page 294) segments by picking one from a pop-up menu on the **Parameter Inspector** (on page 54).

Ø Device Description			
VSL Prime Edition: Strings	Key Switches On:	Off: (optional) Latency (	ms):
Legato Legato s.a., s.r.	F#0C#1F#1	4	Sustained
Legato + soft r. Legato soft att. Legato normal a.	Controller Number: ON Value: 0 0	OFF Value: Latency (	ms):
Legato n.a., s.r. Sforzato Sfz XF tremolo Tremolo	Program Change 1	Latency (	ms):
Trem marcato Pizzicato+marcato Pizz	Overlap		
Portam.+norm.a. Portamento+marc Portam.+s.a., s.r.	1/16 ❤ Default		
Portam.+n.a., s.r. Portamento Portam.+soft r.	lass out Function Man		
+ - Portam.+s.a., s.r.	Import Expression Map		
OK Cop	y Paste	Clear	Cancel

An articulation is transmitted as a **Key Switch** (KS), **MIDI Controller** (CC) or program change message. Please consult the documentation of the sound library, plug-in or hardware for details.

During playback, an articulation is turned on shortly before the start of the Figure segment or symbol. With the **Latency** setting you can determine how much ahead of time an articulation is sent. The default is a few milliseconds but your sound library may require more headroom for a sound to switch. Note that this setting is equal for all articulations of the same type.

An active articulation is sustained until a subsequent Figure segment either requests a different articulation, or has no articulation assigned to it at all, in which case Synfire switches back to the **Default** articulation. It is thus important to define one articulation as **Default**.

If a sound requires that every articulation be turned off immediately after use, you can set an optional Off value.

Articulations such as portamento and legato can be provided with an **Overlap**. The notes of a Figure segment are then output overlapping so that the instrument can generate a corresponding transition. This has the advantage that you do not have to adjust the lengths by hand and they still correspond to the notated notes.

# CAUTION:

If different articulations for the same sound on the same MIDI channel overlap in time, they may interfere with each other. The same holds for multiple voices within the same **Figure**.

# Inheritance

Custom Articulations can be defined per <u>Device (on page 36)</u>, per <u>Sound (on page 37)</u> and per <u>Instrument (on page 33)</u>. And they are inherited in that order: An articulation defined for a **Device** is automatically available to all its **Sounds**,

which in turn is available to all **Instruments** that are using the sound. An individual sound or instrument may add to or override the inherited definitions.

This inheritance is accessible through tabs on the configuration window.

# **Import Expression Maps**

Synfire can import **Cubase Expression Maps** and thus save you the effort of adding individual articulations manually.

It is important to keep in mind that articulations are identified by name when you assign them to a **Figure** segment or symbol. Changing an articulation's name in a device description does not automatically update the Figures in all your arrangements. We recommend you don't change any imported names. This will also keep your arrangements compatible with those of other users that use the same sound library.

# **Copy To Sound**

When this switch is enabled while you are editing the settings of an **Instrument**, the settings will also be written through to the **Sound** where they are saved permanently along with the Device Description. This is useful if you want to preserve your incremental edits for later reuse by saving a **Rack Module Preset** later on.



This feature is available with the Pro edition.

# **Online Updates**

Synfire will show you a notice when an update is available on our website. With *Help > Online Updates* you can examine the type of update that is available. There are two types of updates.

## **Updates**

Updates always include a full program installer. At this time Synfire cannot yet update itself, so you need to login to you user account and download it from there. Future versions of Synfire might make this procedure more convenient and integrated.

## **Patches**

When a release is basically stable, minor bug fixes and improvements can be deployed as patches. Many patches can load into Synfire while it is running. So most of the time you can continue working without restarting the program. With patches, the turnaround is much faster, because there is no need to upload, download and install full updates.

Double-click on the update package to select it for download and then click on Download and Install

# **Crash Reports**

A complex software like Synfire is never really finished. It is constantly being worked on, be it to fix bugs, improve performance or add new features. You can help facilitate and speed up this process by submitting occasional crash reports to us.

If you experience an **Unhandled Exception** error, the current state of the program at the moment of the crash and the text of the Synfire console are collected in a file on your computer (runtime-errors.dump).

Open the dialog *Help > Online Updates* and click the button Submit Crash Reports. All reports collected so far will be sent to Cognitone and then deleted from your computer. Our developers work through the list of reports to understand the crashes and provide fixes for the next update or patch.

# Note:

**We value your privacy**. A crash report does not collect information about you, your music or your computer, except basic parameters like operating system version and CPU architecture. Unlike the system reports and crash reports you may know from Apple or Microsoft, no detailed information about your hardware or other installed software is included. The dumps are encrypted merely to prevent hackers from learning about potential security vulnerabilities of the program. Reports are deleted when they are no longer needed or their version of Synfire became outdated.

# **Downloading Patches On Other Computers**

In case your music computer doesn't have access to the Internet, you can download patches on another computer and move the file to your music computer for installation. Save the downloaded files to a portable medium with *File > Export Updates*. Then install them on your music computer with *File > Import Updates*.



Important:

You can only install patches you have downloaded from your own user account.

# Import MIDI File Dialog

Before you begin, make sure you understand the limitations of MIDI import (on page 264).

# **Tracks**

Select and configure the tracks you want import on the **Tracks** tab. The number in square brackets indicates the MIDI channel. Tracks that should be imported appear in bold. An  $\frac{1}{2}$  means the track will be considered for harmonic analysis.

	Tempo Sig 208 4:	nature Offse 4 2m	et Length 132m	Key C.Major			?
	Tracks	Settings	Progress				
Import All Tracks as Static Pitches (Fast No.   Channel   Harmonic   Track Name: 1 [1] H Trumpet 2 [3] H Acoustic Bass 3 [2] H Yamaha Grand Piano		All Drur Selecte 1237 not Vour Cho V Impo V Harn Sound: D Samu Global Ra FluidR3 / Piano [En	ns d Track es, 141% of averag ice: rt Track nonic Content e For All Tracks ck / Soundcase Yamaha Grand gine100 Tchannel	Figure R Preset Keyboar Algorithm Pattern 1/64 ↓ 3 ↓ V Trans V Susta Simpl Simpl	ecognition ds (rhythmic) Recognit Resolution Separate pose ined ify Bass ify Chords	Symbols ✓ Vertical ✓ Horizontal ✓ Chords ✓ Bass Accidentals Ranges Debug	V Auto

#### Import All Tracks As Static Pitches (Fast)

Configures all tracks to skip **Figure Recognition**. This is the fastest way to import a file. After the import has finished, you can go through each imported track or phrase and run the **Harmonizer** and **Figure Recognition** more selectively as needed on the <u>Take (on page 312)</u> parameter.

## **All Drums**

Force all instruments to use a drum kit sound. By default this uses the global **Drums** instrument, but you may assign a different drum kit to any track, which will then be used instead. This is especially useful for batch import.

#### **Import Track**

Check this for each track you want to import.

#### **Harmonic Content**

Check this for each track that contains tonal music (pitched instruments). The more the better. Leave unchecked for drums and percussion.

#### Sound

Select a sound for the track to use after import. This influences the selection of a default <u>Interpretation</u> (on page 301) and suggests **Playing Ranges** the MIDI track should be transposed to in order to match. If the desired sound its not yet available, you can use the <u>Sound Wizard (on page 193)</u> as usual to create one.

# **Same For All Tracks**

Use the selected sound for all tracks and all MIDI files. This makes sense if you import a batch of files with phrases for a single known instrument.

## **Figure Recognition Settings**

Configure this for each track the same way you would to for recording (on page 74).

# $\Box - \triangleright - \mathscr{P}$

Preview one or all tracks before import.

# **Settings**

Configure pre-processing and post-processing on the Settings tab.



# **Import Entire Directory**

In addition to the file you are currently setting up, Synfire will also import all other MIDI files it can find in the same directory.



## **Include Subfolders**

Whether to include all subfolders recursively with a batch import.

## Merge All Tracks Into One

Merged all tracks into a single track before import. Good for drum patterns.

#### **Identical Track Layout**

All files contain the same instruments in the same order. Your settings per track are adopted as far as possible. Files with a different number of tracks are skipped.

#### **Create a Library Folder Per File**

Adds a folder to the library for each file that contains more than a single track with content. This groups imported tracks per file. You should keep this checked unless you want to merge all imported tracks in the same folder.

#### **Labeling Of Phrase Pools**

Select the components to use for labeling each imported phrase pool.

## **General MIDI Standard**

If enabled, this will interpret the program change numbers found in the file according to the GM Standard and assume the instrument category and sounds associated with that.

#### **Retain MIDI Channels**

If enabled, an attempt is made to allocate the imported sounds on the very same MIDI channels that were found in the file. This will probably not work with the default sounds on the global rack, because many channels are already blocked by **Global Instruments**. You should therefore create a dedicated multi-timbral rack module with the **Sound Wizard** (*on page 193*) and assign the MIDI tracks exclusively to sounds provided by it.

#### **Separate Channels For Equal Sounds**

If the same sound is used on multiple MIDI channels (tracks), the sounds are probably supposed to be separate of each other for independent control and mixing. This separation may demand more dynamic channels than available. If your are importing MIDI files with more than 16 tracks, you may need to disable this.

#### **Import Controllers**

Import MIDI controllers to the corresponding parameters, if supported.

#### **Tempo Changes**

Import tempo changes to the Tempo (on page 313) parameter.

## **Disable Automatic Transposition**

Tracks are transposed by octaves to ensure they meet the destination instrument's total playing range. This is vital for proper figure recognition. If you disable this, the playing range of the instrument is transposed instead.
#### **Create Figures**

Disable this if you want to import <u>Harmony (on page 300)</u> and <u>Take (on page 312)</u> only and do the <u>Figure Recognition (on page 5)</u> later when sifting through the phrases.

#### **Create Phrase Pools**

(Library only) Create a <u>Phrase Pool (on page 28)</u> for every track with multiple phrases extracted automatically based on content. Duplicate phrases are dropped.

#### Average Phrase Length (Menu)

Select a desired length you want the average phrase to have. Actual lengths depend on the content found on each track though.

#### Simple

Divide all tracks into sections of this length without paying attention to the content.

#### **Keep Takes**

Disable this if the file is huge and you want to save memory in the target arrangement. It is a good idea however, to always keep the <u>Take (on page 312)</u>, as you can re-run **Figure Recognition** and **Harmonizer** on it at any time.

#### **Keep Master Take**

The take that includes an entire track's worth of MIDI data is called the Master Take. If you already keep individual takes per phrase, you can drop the Master Take to save a lot of memory in the library.

#### **Detect Keys, Chords, and Scales**

Whether to create a <u>Harmony (on page 300)</u> parameter from all tracks with tonal content. Disable only for drum patterns.

#### **Use Key Read From MIDI File**

If the file contains key signature information, it will be considered (not all MIDI files contain this information)

#### **Allow Key Changes**

Recognize and import key changes. Importing files with only a single key is somewhat more robust.

#### **Confirm Detected Key**

Ask the user to confirm the detected key. You will have the option to select among the most likely keys yourself. Knowing the key is vital to harmonization.

#### **Complete Chords**

If the MIDI file contains sustained chords in root position, you can use this option to import the chords directly and unchanged as progressions. All intelligent algorithms for analyzing non-quantized recordings are then deactivated.

#### **Prefer Orchestral Interpretations**

Where multiple Interpretation settings may apply, Synfire will prefer an orchestral Interpretation over equivalent others. Enable this if you are importing classical music files.

#### **Starting The Import**

Press **Start Import** to start the process. Depending on the complexity of the file, this process can last for a long time. After import has finished, a new arrangement or library will open with the result.

#### **Progress**

On this tab you can follow the import process, which may take quite a while. You may terminate the process at any time with **Cancel**.

			Tempo	Signature	Offset	Length	Key	
			90	4:4	1m	31m	A.Minor	
	Tracka	Sottings	Drograce					
	TIACKS	Settings	Flogress					
Eviting								
Entering Polyph	onic Patteri	n Recognitior	า					
First look shows	4 areas of	interest at C	)  1/4   67/4	+1/8+1/16   83	8/4+1/8+1/′	16		
Recognized 7 Cl	hord Segme	ents						
Recognized 4 R	rills	lies						
Recognized 777	Melodic Se	egments						
Recognized 12 s	symmetrical	l shapes						
Considering 808	3 hypothese	es in 6 distind	ct clusters					
Entering Linear	Segment So	canner						
First look shows	8 areas of	interest at C	)   1/4   26/4	39/4   44/4	59/4   8	37/4   92/4+	1/8	
Recognized 32 I	Repetitive L	ines						
Recognized 39 Recognized 100	I rills 9 Melodic 9	Seamente						
Recognized 4 sv	vmmetrical	shapes						
Recognized 5 sy	, /mmetrical	shapes						
Recognized 4 sy	ymmetrical	shapes						
Recognized 1 sy	mmetrical :	shapes						
Considering 110	2 hypothes	es in 9 distin	ct clusters					
<u> </u>								
						C.	ncol	

### **Batch Import Of Folders**

You have two options to import multiple MIDI files as a batch.

- 1. Import a single MIDI file with File > Import > Standard MIDI File ....
- 2. Import a folder with multiple files with File > Import > Standard MIDI Files (Folder) ....

Since you can only adjust settings for a single file, the same settings will be used for all files. Although figure recognition settings need to be auto-detected for each individual track, because it is unlikely that all files share the same ordering of instruments and types of content. Batch import therefore works best for drum patterns and high-quality MIDI files that respond well to auto-detection.

i Tip:

If you import drum patterns, be sure to check Import All Tracks As Static Pitches (Fast).

# 👔 Tip:

After a large batch import, don't save the huge imported library directly. Instead, open another Library window and move over only those phrases that you want to keep. Saving huge libraries takes a long time and if you only want to keep 20% of its content anyway, this is a much more efficient workflow.

# **Keyboard Shortcuts**

With *Edit > Preferences ... > User Interface > Keyboard Shortcuts* you can remap almost all commands to the keyboard as you see fit.

D Global							
▷ Application Windows							
	ture						
Container							
New Contai	ner			ሪኑ	€N	Мар	Reset
Rename						Мар	Reset
Pack						Мар	Reset
Close Gap						Мар	Reset
Make Room						Мар	Reset
Inactive						Мар	Reset
Priority Up					↑	Мар	Reset
Priority Dov					$\downarrow$	Мар	Reset
Keep Child	Containers in a Row					Мар	Reset
Make Alias	Make Alias					Мар	Reset
Make Alias	Physical					Мар	Reset
Make Sketc						Мар	Reset
Select Origi	nal					Мар	Reset
Make Snaps	shot					Мар	Reset
Loop Conta	iner					Мар	Reset
Loop On/Of						Мар	Reset
Render						Мар	Reset
Selection							
Move Right					→	Мар	Reset
Move Right	(fine)			¢	> →	Мар	Reset
Move Left					÷	Мар	Reset
Move Left (	fine)			¢	÷←	Мар	Reset
Stretch							
Shorten				<b>۲</b> ₩	*←	Map	Reset
Reset to	Default	Load	нтмі		Cancel		ОК
Reset to	bondan	Load			Cancer		

D

Expand or collapse sections of the list. Hold down **#** Control to collapse or expand all at once.

#### Мар

Click to remap the command. The label flashes while Synfire is listening for keyboard input. A warning will show if the command is already mapped to another command.

#### Reset

Reset to the system default mapping.

#### **Reset to Default**

Reset all commands to the system default mappings.

#### Load

Load mappings from disk.

#### HTML

Save a HTML page documenting all mappings that you can view or print with your web browser.

#### Cancel

Changes are not saved to disk.

#### ΟΚ

Changes are saved to disk.

# Drone

### The Cognitone Drone audio plug-in

The **Cognitone Drone** is a VST, VST3 or AudioUnit plug-in that you can load into a DAW. It should appear in the plug-in browsers and menus of you DAW, unless you have installed it to a directory the DAW is not looking into when it scans for plug-ins (Windows).



A Drone hosts audio plug-ins on behalf of Synfire and feeds them with a MIDI stream ahead of time.

#### **Numbered Colored Area**

The number of the drone, which is also part of its port name. This number is assigned automatically. Next to the number you see the name and type and edit status of the guest plug-in that is currently loaded (asterisk = has been changed). The color indicates the status of the drone (see Error: Reference source not found on page Error: Reference source not found).

#### **Big Title**

Name of the device description you have connected to the port of the drone. By choosing a suitable name for the device, you can easily recognize which instruments of your arrangement are hosted in this drone.

#### **Small Text**

Shows the name of the arrangement this drone belongs to, as well as additional information as needed.

#### **Playback Mode (Menu)**

**Play DAW**: The Drone plays local MIDI data from the track in the DAW only. It ignores any music streamed from Synfire. This is useful when you already exported MIDI to your DAW and want to keep the guest plug-in inside the Drone.

Play Synfire: Only MIDI streamed from Synfire is played. Data on the DAW track is ignored.

Play Both: Both data received from Synfire and the DAW track will be merged and played together.

#### Export

Indicates that the entire MIDI content rendered by Synfire is present in the drone. You can drag it off at the colored area and drop it into a DAW track to export the generated music.

#### Online

The drone is connected to Synfire.

#### Load

Load a VST/AU instrument into the drone manually. You should avoid this and load plug-ins from the menus and browsers in Synfire instead.

#### Editor

Open the guest plug-in editor. When making changes to the guest plug-in, please note that saving the DAW project alone is not sufficient. All changes also need to be saved by Synfire. This is automatically done each time you save a Synfire project or the global rack.

#### Troubleshooting:

Should a particular plug-in cause problems with the editor, you can try opening it in a separate window: Hold down the SHIFT key while pressing the Editor button. The drone retains this setting until you undo it by making the selection again.

#### Note:

1

This feature is available with the Express and Pro editions.

# **Chapter 3. Factory Documentation**

# **Documentation of KIM Factories**

This section lists all currently available factories and explains their purpose and usage.

Note: This feature is available with the **Pro** edition.

### **Related information**

Factories (on page 29)

# Elements

### List of common elements

There are a few elements that occur as a part of many factories. These are explained here in detail.

#### **Related information**

Factories (on page 29)

# **Metrics**

Set the metric **Scheme** (time signature), **Tempo** and number of measures to generate. These are global per phrase and influence all factories involved with generating it.



# Note:

If you generate a phrase with a different time signature, this will start a new phrase pool.

### **Related information**

Factories (on page 29)

# Similarity

These sliders control how similar something is expected to be to something generated earlier in the phrase. On the extreme, the **Identical** setting will repeat previous output. On the other extreme, the **None** setting leads to distinct and unique output without any repetition.



#### **Intra Measure**

How similar things are supposed to be within the same measure (short range).

#### **Beyond**

How similar things are supposed to be in subsequent measures to what has already happened in previous measures (long range).

#### **Related information**

Factories (on page 29)

# Spans

Using these probabilities as a guideline, a factory first generates empty spans according to its internal rules. After spans have been generated, they are assigned a <u>Type (on page 215)</u> of Figure segment and filled accordingly. Longer spans allow Figure segments to run and develop, while very short spans may clip them to a blip.



The setting **Crossing Measures** allows spans to overlap measure bounds, which is good for melodies and weakens the pronunciation of beat.



The lengths shown here are merely an example for 4:4 (common time). Lengths for other time signatures may vary.

#### **Related information**

Types (on page 215) Factories (on page 29)

# Types

The <u>spans (on page 214)</u> generated by a Factory are filled with Figure <u>Segments (on page 294)</u>. These probabilities control which types of segments are supposed to occur.

Types						
Segments		21%				
Arpeggios		14%				
Octaves		51%				
Ostinato	• • • • •	7%				
Ornaments	• [] ]	7%				
Glissandos		0%				
Variations						
Variations	<b></b>	0%				
Stacking	mu	0%				
Interval	3					

#### Segments

Segments that are controlled by the settings you make on the <u>Segments (on page 217)</u> tab. These are mostly melodic runs.

### Arpeggio

Segments running up and down the line grid in a triangle-like shape. Depending on the symbol type, you get an arpeggio for chord symbols or a melodic run for vertical or horizontal symbols.

#### **Octaves**

Segments alternating between 0 and 7, which however is not necessarily always an octave, depending on the current scale.

#### Ostinato

Segments repeating the same symbol multiple times.

#### Ornaments

Grace notes or trills.

#### Glissandos

Segments gliding up or down the grid very quickly.

### Variations

#### Variations

Controls how likely individual segments are transformed, e.g. flipped upside down or reversed.

#### Stacking

Controls how likely two copies of a segment are stacked above each other.

#### Interval

The scale steps between stacked segments.

#### Important:

How many symbols are actually filled into a span depends on the **Step** parameter generated internally by the Factory and the length of the span. There may be spans that come up empty, because none of the steps happen to fall into that span. Likewise, there may be spans too short to contain enough of a segment type to make it recognizable as such. That is, you might get only one or two symbols where an arpeggio or melodic run is supposed to appear.

#### **Related information**

Spans (on page 214) Step Distribution (on page 220) Factories (on page 29)

# Segments

Control the **Symbol Type**, **Playing Range** and shape of **Figure** segments to be generated. Configurable number sequences control how a segment is drawn incrementally. The optional breakdown of a segment into **Head** and **Tail** allows the beginning and the further course of a segment to be based on different number sequences.

Some settings only affect spans (on page 214) of the type Segments.

Symbol	Horizontal 🗸					
Playing Range	Default 🗸					
	Enable Triplets					
	🗸 Tails					
Start Heads From Anchor Backwards						
	<b>.</b>					
Starts	0246	Кеер	~	Per 1m	~	
	Heads					
A	2 -1	Кеер	~	:   Last 1	~	
В	-2 1	Rotate	~	:11	~	
С	3 - 2 1	Rotate	~	:   Last 2	~	
	Tails					
A	-101	Rotate	~	:11	~	
В	001-12-2	Read 3	~	:   Last 1	~	
С	0 0 1 -1 2 -2 3 -3	Permute	~	:   Last 2	~	

#### **Symbol**

The type of Figure Symbol to use.

#### **Playing Range**

The Playing Range to assign to the segment being drawn.

#### **Enable Triplets**

Allow individual segments to advance in triplets as a random variation. If you need more triplets or other tuplets, paste the desired rhythm into the **Parameter Outlet** of the **Step** factory.

#### **Start Heads From Anchor Backwards**

If enabled, the current value of **Starts** controls on which grid line the anchor is drawn and the **Heads** sequence is glued to it on the left side in backwards direction.

#### **Starts**

Determines the position on the line grid where the drawing of a new segment starts.

### **Heads**

The beginning of a segment is called the **Head**. It is drawn first. These numbers determine by how many steps the writing head is transposed up or down **after** a new symbol has been drawn. If **Tails** are activated, the heads are drawn in the reverse direction, starting from the anchor to the left..

#### Tails

The further course of a segment is referred to as the **Tail**. If tails are not activated, segments only consist of a **Head**.

### **Number Sequences**

All numbers denote vertical moves up or down the current scale, while horizontal movement is exclusively controlled by the <u>Step (on page 220)</u> parameter. That is, you have a rhythm beforehand and the sequences make a melody move up or down while the current position advances to the next step with each new number.

Every number sequence can be configured to deliver a random variation of itself and how it is looped. The input field takes a list of one or more numbers separated by space or comma. The two menus to the right of it control **Transformation** and **Streaming** of the sequence.

### **Transformation**

Depending on this setting the sequence delivers a random variation every time it is used by the factory. The randomness is controlled by the following settings.

Кеер

Takes the elements unchanged as they are (no randomness).

#### Permute

Delivers a random reordering of all elements.

#### Rotate

Scrolls around the sequence of elements by a random number of steps.

#### Read 2, 3, 4

Selects 2, 3 or 4 successive elements starting from a random position, while keeping their order.

#### Pick 1, 2, 3, 4

Picks 1, 2, 3 or 4 elements randomly in a random order.

#### Streaming

Controls how a sequence is looped or stretched and interpolated evenly to cover a certain span of time.

Once

Streams all elements once and stops at the end.

:||

Repeats the sequence of elements indefinitely.

#### :|| Last 1, 2, 3, 4

Streams all elements and then repeats the last 1, 2, 3 or 4 elements indefinitely.

#### **Per Phrase**

Evenly distributes all elements over the entire span of the phrase.

#### Per 1m

Evenly distributes all elements over each new measure that is generated.

#### Interpolated

Inserts interpolated values between the evenly distributed elements. For example, the sequence 0 9 with *Per Phrase* streaming and *Interpolated* will stream 0 1 2 3 4 5 6 7 8 9 evenly distributed across the phrase.

# i Tip:

A vast range of melodic shapes can be generated from these configurable sequences. You should keep in mind though, that this is for a single line of one phrase only. Nothing stops you from generating many similar variations using the same factory and then drop them on a phrase in your arrangement at different positions in order to glue together a slowly evolving theme.



If you want only a single sequence, simply configure A, B and C identically.

### **Related information**

Step Distribution (on page 220) Spans (on page 214) Factories (on page 29)

# **Step Distribution**

A Factory generates a **Step** parameter internally that is used to determine the onsets and lengths of all generated **Figure** symbols. It is the rhythmic grid for all output. Symbols always sound until the next step in the grid (no gaps). The only way to modulate symbol lengths is the **Flow** parameter.

These probabilities are used by a Factory's rules, which take into account a lot more that simple random distributions in order to build a musically meaningful rhythm. Pulling either slider to 100% will give you a constant step.

Step distribution may also be used by Factories that generate parameters other than Figure.



The **Dotted** slider controls how likely these note lengths are extended by half their duration.

# Note:

The lengths shown here are merely an example for 4:4 (common time). Lengths for other time signatures may vary. The meaning of 1/1 may be synonymous with 1m where time signature is not a multiple of 4:4.

### Note:

Triplets and quintuplets only make sense as multiple steps in a row. They can't be arbitrarily inserted anywhere. Hence they are taken care of by the settings that control the generation of Figure Segments (on page 217) and must be enabled on that tab.

# Note:

Lengths shorter than 1/32 are used by specific segments types internally only, like Glissando and Ornaments.

#### **Related information**

Spans (on page 214) Types (on page 215) Factories (on page 29)

# **Beat Divisions**

Control how likely a value is expected to be set for a parameter in different zones of a metric scheme.



#### **Start of Row**

The very first beat of a phrase or progression.

#### **Down-Beat**

The first beat of every measure (bar).

#### **On-Beat**

Every heavy beat.

#### **Between Beats**

Any position that is not close to a beat.

#### **Off-Beat**

Every light beat.

### **Up-Beat**

The last light beat of a measure, before a new measure starts.

#### **Related information**

Factories (on page 29)

# **Parameter Factories**

# List of parameter factories

#### **Related information**

Factories (on page 29)

# **Rhythmic Bass**

### Flow: Generate articulations typical for acoustic bass

Generates a <u>Flow (on page 298)</u> parameter that stops notes at rhythmical positions as is typical for acoustic bass. Note that this parameter is used only while data is being generated. It is not added to the generated phrase as a visible parameter.

# **Metric Stops**

### Flow: Generate stops and other flow values based on beat zones

Generates a <u>Flow (on page 298)</u> parameter that shortens note lengths for a stop effect at rhythmical positions based on probabilities.

### **Off Value**

Default where no value is applied.

#### **On Value**

Value to apply at beat positions as determined by the probabilities below.

#### **On Span**

Span to fill with the On Value, starting from a beat.

#### Invert On vs. Off

Flips entire parameter upside down.

#### **Rotate Left**

How much to shift the span in order to take effect earlier.

#### Period

Loop length of the parameter.

# **Rhythmic Chords**

### Flow: Modulates note length based on time signature

Factory that modulates note length based on time signature, best for rhythmic chords.

# Static

Flow: Provides a constant flow value

This factory forces a constant flow setting on all notes.

# Functional

**Harmony**: Generate random chord progressions based on functional relationships and popular cadences.

Factory for random chord progressions based on functional relationships and popular cadences.

# **Chord Machine**

Harmony: Generate random progressions unfolding along a probabilistic chord network

Harmony	
ł	larmony
	Metrics
Chords	4 🗸
Signature	4:4 🗸
Progress	Slow (1m) 🗸 🗸
	Material
Key	A.Major 🗸
Richness	Medium 🗸
	Re-Assess Key, Scales
	Options
	V-I Ending
Sec. Dominants	None 🗸

Factory for random progressions unfolding along a probabilistic chord network.

# Stray Chords

# Harmony: Extensive factory for chord progressions

Generate otherworldly and unique progressions with up to two selectable keys and predefined start or end chords to match. This is an experimental, yet very powerful factory that will provide you with an endless supply of inspiring chord progressions. Here are a few tips for adjusting the settings.

Harmony		
Harmony	Constraints	Preferences
Metrics Chords 8 ~ Signature 4:4 ~ Progress Slow (1m) ~	First Chord Last Chord V7 A Fixed Notes i Moves -7 Move Along Scale	Key Membership    6%      Harmomic Functions    6%      Similar Size    37%      Avoid Repetition    35%      Known Changes    100%
Material Key-less	B Fixed Notes iv Moves -1 5 -7 Move Along Scale	Strong Changes IIII 2% Chromatic Moves IIII 28%
Key 1 A.Major Key 2 F.Major Richness Open Catalog Priority 5 Re-Assess Key, Scales	Carry Notes 2 V Carry Steps 1 V	

#### Looped

The transition between last and first chord is optimized for loops.

#### **Key-less**

Disables the focus on any key signature for free roaming chord progressions.

### Key 1, Key 2

The first key is considered the main key, while an optional second key provides additional chord material. Set both to the same key if you don't need a secondary key.

### **Richness**

- Simple: Up to four notes and favors triads.
- Medium: Up to five notes and includes diminished chords.
- Complex: Up to five notes and includes diminished and augmented chords.
- **Open**: Up to six notes of all chord types and relaxed constraint checking.

#### **Catalog Priority**

Prefer chord structures with a priority in the Catalog equal to or lower than this.

#### **Re-Asses Keys, Scales**

After generating the whole progression, apply the keys and scales estimation algorithm. Disable this if you want to keep the keys assigned by the factory.

#### First, Last Chord

You can predefine a first and/or last chord to have the factory fill everything in between. Input may be chord names or **Roman Numeral** expressions.

#### **Fixed Notes A, B with Moves**

You can specify that each chord must contain one or more fixed notes. Enter them with names or **Roman Numerals**. If you enter a sequence of numbers in the "Moves" field, the respective next chord will be transposed accordingly. Transposition is chromatic by default, but can be switched so that it occurs along scale steps.

#### **Carry Notes, Steps**

You can require a number of notes to be carried (sustained) across chord changes for a minimum number of steps.

#### Preferences

Multiple sliders allow for adjusting the priority of various properties the chords and transitions between them are supposed to have.

# 🚹 DANGER:

This factory is experimental. Some settings may lead to constraints that are impossible to meet and that may possibly hang the algorithm.

### **Metric Pauses**

### Pause: Drop notes based on rhythmical positions

Generates a <u>Pause (on page 306)</u> parameter that drops notes based on rhythmical positions. Anywhere this parameter has an "Off" value, figure symbols or parameter data is filtered out.

#### **Pause Span**

How long, starting with the beat, a pause will last.

#### Invert On vs. Off

Flips the resulting parameter upside down.

#### **Rotate Left**

How much the pause span will be shifted early to the beat.

#### Period

Loop length of the parameter.

Note that this parameter is used only while data is being generated. It is not added to the generated phrase as a visible parameter.

# Alternating

### Skip: Skip positions of the Step parameter based on rhythmical positions

Factory for skipping steps for multiple melodic lines

# Coordinated

### Skip: Provides three coordinated parameters

Factory provides three coordinated Skip parameters

# Bass One

### Step: Generate steps based on patterns typical for acoustic bass

Generates a <u>Step (on page 310)</u> parameter based on patterns that are typical for acoustic bass. Also very good for other instruments that take on a bass role.

# Random (Flat)

#### Step: Generate random steps

Populates a loop with random steps.

#### Period

Loop length of the parameter. Longer periods will make the resulting rhythm appear more random and chaotic, as there is probably no discernible pattern.

#### **Preserve Down-Beats**

Make sure there is a step at every down-beat at the beginning of a new measure.

# Sequencer

### Step: Provide a sequence of incremental steps manually

Generates a <u>Step (on page 310)</u> parameter allowing you to input your own steps and loop length. For details on sequence entry please <u>look here (on page 219)</u>.

# Random (Structured)

#### Step: Generate steps with structure and self-similarity

Generates a <u>Step (on page 310)</u> parameter with deep structure and adjustable self-similarity for natural sounding rhythms.

### Syncopes

### Step: Generate steps with all on-beats removed

Generates a Step (on page 310) parameter for rhythm patterns that have all on-beats removed for a syncopated feel.

# **Metric Velocities**

#### Velocity: Generate curves based on time signature

Generates velocity curves based on time signature.

#### Invert (Emphasize Off-beat)

The selected waveform emphasizes on-beats by default. This inverts the waveform to emphasize offbeats.

#### **Half-Measure Dip**

Makes the curve less pronounced in the middle of a measure, putting more emphasis at the beginning.

#### **Double Speed**

Doubles the frequency of the waveform.

#### Push

Adds extra emphasis early to the beat to make it more pronounced.

#### Rotate

Shifts the wave by a set amount, making dynamics feel early or late to the beat.

#### Humanize

Adds random variation in dynamics.

#### Polyphony

Generates multiple velocity curves, so chords can obtain a different value for each note.

# Static Velocity

### Velocity: Provide a constant velocity

Factory that forces a constant velocity on all notes.

# **Line Factories**

### List of line factories

#### **Related information**

Factories (on page 29)

# Bass (Multi)

Bass lines based on three separate sets of heads and tails

This bass factory is based on three separate sets of heads and tails for additional variation.

# Bass (Single)

### Bass lines based on a single set of heads and tails

Configurable bass factory based on a single set of heads and tails.

# **Piano Bass**

# Bass lines typical for piano left hand

Bass factory for what is typically played by a piano left hand. Also very good for other bass instruments.

# Arpeggio Chords

Generate broken-up arpeggio chords

Useful as a replacement for a chords part. Also great for other symbol types.

# **Guitar Chords**

Strummed guitar chords

Configurable factory for strumming guitar chords.

# **Piano Chords**

### Chord stabs for piano and other keyboards

Configurable factory for chord patterns. Useful for piano and other keyboard instruments

# Arp One

### Configurable arpeggiator with many options for rhythm, pauses and dynamics

Configurable arpeggio factory with many options to control rhythm, pauses and dynamics

# **Chained Elements**

### Sequentially chained melodic elements

Factory for sequentially chained melodic elements. Good for solo and improvisation

# Generic Melody (Single)

Melodies based on a single set of heads and tails

Configurable melody factory with a single set of heads and tails.

# Generic Melody (Multi)

### Melodies based on multiple sets of heads and tails

Configurable melody factory with extra variety offered by multiple sets of heads and tails.

# **Piano Right Hand**

# Melodies typical for piano right hand

Melody factory typical for piano right hand, but also great for other purposes.

# Question + Answer

### Interleaved melodies following the question and answer idiom

Melody factory with two separate sets of parameters, interleaved for a question and answer feel.

# **Develop A**

### Generates longer melodies that evolve over time.

Generates a melodic line that develops variations and changes over time. It uses two steps. A slow **Anchor Step** is used to place anchors, while a faster **Melody Step** is used to place individual notes. Two variations of the latter are generated internally and you can decide how frequently they will alternate.

# Vocals

### Generates melodies useful for vocals.

Generates a melodic line that resembles characteristics of lead vocals. It uses two steps. A **Slow Step** is used for long notes, while a **Fast Step** is used for shorter notes.

# **Phrase Factories**

### List of phrase factories

#### **Related information**

Factories (on page 29)

# **Counterpoint Melodies**

### Three freely configurable lines

Factory for three configurable lines that follow individual step rhythms. Useful for splitting across multiple instruments.

# i Tip:

Like most factories, this one can be used for much more (on page 32) than its name suggests.

# Counterpoint (Skipped)

#### Three lines based on a common Step, each using a different Skip

Factory for three configurable melodic lines, each using a different Skip parameter for rhythmic effects.



Like most factories, this one can be used for much more (on page 32) than its name suggests.

# **Piano Parts**

# Left hand, right hand and chords, great also as three lines for different instruments

Factory for three parts: Left hand, right hand and chords. While this was originally designed for piano, its is great for any keyboards instrument, or for splitting the three parts across multiple instruments.

# 7 Tip:

Like most factories, this one can be used for much more (on page 32) than its name suggests.

# ExperimentalKit1

# Experimental drums factory based on a configurable line per instrument

Experimental factory based on a configurable line per instrument. Good for break beats and a starting point for manual refinement and inspiration.

# Chapter 4. Tasks

Step-by-step instructions of common tasks

# Audio Setup

Setting up audio drivers, devices, sounds, racks and drones

Video Tutorial

# **Assigning Sounds**

#### Video Tutorial

There are multiple ways to select a Sound (on page 37) for an Instrument (on page 33).

#### **Rack Module Library**

Search and browse the <u>Rack Module Library (on page 47)</u> on the left sidebar that contains rack module presets, plug-ins, MIDI ports, Drones and global instruments.

Drop any available item on a <u>Track (on page 20)</u> to create a **Rack Module** and select one of its sounds.

Drop items below the bottom track to append a new track.

ß

Anywhere you see this icon, you can open the <u>Sound Wizard (on page 193)</u>. Alternatively you can open it from the menu with *Instrument > Assign Sound* .... It guides you through the process of setting up or extending a **Rack Module** and creates a **Device Description** where one is needed, so you don't have to. This method is best when you need to add new sounds to an arrangement.

#### 夏

This icon shows up on tabs related to sound assignment, for example the **Parameter Inspector** on the right sidebar. There you can select an existing **<u>Rack Module</u>** (*on page 38*) and channel from pop-up menus to use a sound provided by that module. This method is convenient for selecting fixed-channel sounds you have already added to an arrangement.

#### **Copy From Other Arrangements**

Drop any <u>Rack Module (on page 38)</u> from another arrangement's rack on your <u>Arrangement Rack (on page 24)</u>. Sounds are then available for the <u>Sound Wizard (on page 193)</u> to browse.

#### **Manual Setup**

Visit the <u>Arrangement Rack (on page 24)</u> and add a new <u>Rack Module (on page 38)</u> and set it up as <u>explained here (on page 232)</u>. Sounds are then available for the <u>Sound Wizard (on page 193)</u> to browse.

#### Important:

When you select a different sound for an instrument, it adopts the sound's <u>Playing Ranges (on page 35)</u> by default. This will likely also change its current MIDI output. You can disable this by deactivating the chain link on the wizard's last page, or globally with *Playback > Audio/MIDI Setup > MIDI > Sound Management*. If you do so, you must verify yourself that your instrument's ranges don't violate the capabilities of the sound.

### **Multiple Instruments Using The Same Sound**

Assigning the same sound to multiple instruments will make them play on the same MIDI channel. If you want one or more of them to play on different channels, you need to multi-select the desired instruments and do *Instrument > Demand Separate Channel*.

### Note:

This only works for devices that select sounds via MIDI program change messages on more than one dynamic channel. Sounds on fixed channels, as the name implies, are always tied to the same channel.

#### **Related information**

Dynamic Sound Allocation (on page 42)

# Setting Up MIDI Input

How to make sure Synfire receives MIDI input from your external keyboard or remote software.

- 1. Open *Playback > Audio/MIDI Setup* and visit the **Inputs** page.
- 2. Find the port of your MIDI keyboard or external interface.
- 3. Check the Input box for it.
- 4. Play notes on the keyboard. The MIDI icon on the window toolbar should flash.
- 5. Also verify the other Input Preferences match your current preference.

#### Troubleshooting:

You may need to do *Playback > Reset Audio/MIDI System* to make recently connected MIDI equipment appear in the list.

#### Troubleshooting:

On Microsoft Windows, problems may occur if multiple programs attempt to use the same USB MIDI driver at the same time. Especially prone to this are cheap drivers that often accompany inexpensive MIDI hardware. If you wonder why Synfire is unable to receive data from your hardware, there is probably another program already using the USB driver. In order for Synfire and other software to receive MIDI at the same time, you may need to use separate USB interfaces, or look for a more capable driver, if possible. Since this is a frequent problem, you should visit the user forum for the latest solutions (users.cognitone.com)

#### Troubleshooting:

On Microsoft Windows, you may experience problems when multiple programs attempt to access the same ASIO driver. In order to be able to run the engines and a DAW in parallel, you might need to assign them to different audio output drivers. For example, you may want to use the ASIO driver for your DAW and the Windows Audio or DirectX drivers for the engines. On the Mac, running multiple Engines and DAWs at the same time is not a problem.

#### **Related information**

MIDI Routing (on page 41)

# **Creating Device Descriptions**

Creating a **Device Description** (on page 36) for your most frequently used sounds is a one-time effort that quickly pays off. Having a device ready to browse for sounds (with all meta-data already set up) will save you a lot of distraction in those moments when your best ideas are coming up.

To obtain a device descriptions for your plug-in or external hardware, you basically have the following options:

#### Migrate

If you already have a description you created with an earlier version of a Cognitone software, you can add it to your list with *File > Open And Add* ....

#### **Download From The Online Repository**

On the tab <u>Repository (on page 182)</u> you can browse descriptions other users have uploaded.

#### **Extract From Plug-In**

Some plug-ins support this. It is worth a try (on page 236).

#### Import from a MIDNAM (XML) File

The user community of Avid ProTools® and Digital Performer® created a collection of MIDNAM files for many synthesizers some time ago. Synfire can <u>import these (on page 235)</u>.

#### **Create One Yourself**

Explained in this section (on page 234) in detail.

For synthesizers you tweak on an individual basis per each project anew, you can also do without a device description altogether. Instead you would enter all required meta-data as you go, guided by the **Sound Wizard**, or in the **Instrument Inspector**.

# New Device Description From Scratch

#### Video Tutorial

Before you start, please consult the documentation that came with your hardware or sound library and make sure you understand the **four types of devices** and which one you need:

#### **Fixed Channels (Sampler)**

Sounds each listen to a fixed MIDI channel. For example, an instance of NI Kontakt<sup>™</sup> with assorted sounds loaded that you often use for songwriting. Each sound is told which MIDI channel to listen on. The entire setup is saved as a preset to get you started quickly with new projects.

#### 1-Channel Multiple Programs (Synth)

The device listens to a single MIDI channel. Different programs/patches are selected by sending MIDI messages or preset selection commands. Sometimes multiple sound banks can be selected. Such a device description typically represents a synthesizer's factory presets or a sound bank loaded into the device.

#### 16-Channel Multiple Programs (Multi-Timbral)

Same as above, only there are 16 MIDI channels, each of which is able to select a different program/ patch to be played at the same time. Typically these devices are **General MIDI** sound modules or classical synthesizer emulations.

#### Template

This device is not an actual device. It merely provides a list of sound names associated with their meta-data that you can lookup and insert when you need them (**Category**, **Playing Ranges**, **Custom Controllers**, **Articulations** and program numbers and selection methods). Setting up a **Fixed Channel** type device from a **Device Template** is especially fast and convenient.

After deciding for a device type, open the pop-up menu with 🗹 from the device list or call up its right-click menu to add a new device.

#### Now you setup its

- 1. Properties (on page 135)
- 2. Variants & Channels (on page 137)
- 3. Programs (on page 138).

#### **Related information**

Dynamic Sound Allocation (on page 42)

### Importing Device Descriptions

You may save some effort by starting a new device description not from scratch, but based on an import of similar meta-data in other file formats.

### MIDI XML (MIDNAM)

Cognitone supports the **MIDI XML** standard format adopted by Apple and the MMA (Midi Manufacturers Association) for the labeling of "patch names" (files extension is **midnam**). Among other things, these files describe the MIDI commands for bank selection and the sounds that are available on a particular sound generator.

On the Internet, **midnam** files are available for many classical synthesizers. You can import those with *Option > Import XML (midnam)*. You'll find currently known download links on the <u>Cognitone</u> website.

Conversely, with *Option > Export XML* it is also possible to export devices in XML format and thus share them with other applications. Note however that between import and export, some information not used by Synfire is lost.

# **Text Files (Sound Banks Only)**

On the Programs tab, with *Option > Import Text File* ... you can import the program numbers and sound names of a single **Sound Bank**. In the text file each line must begin with the program change number, followed by the sound name separated by a tab or space. Of course, you can not import a bank this way before you created a device.

### **Cubase Parse Files**

This is still experimental and not yet available. If you need this, feel free to ask our support for assistance.

# **Extracting Plug-ins**

#### Video Tutorial

Some audio plug-ins support automated extraction of their list of programs/patches. Although this capability is part of the VST and AudioUnit specs, many developers do not bother implementing it. In general, it is more likely supported by synthesizer type plug-ins than samplers or sound libraries.

After you started *Extract From Plug-in* ... from a rack module's device menu, Synfire will ask you for a name for the device description about to be extracted.

If for example the plug-in **NI Reaktor** has a patch Lazerbass loaded, you would enter **Lazerbass** here and the device will contain all programs/patches available for Lazerbass.

You also can leave the name empty. Synfire will then guess a name based on what it finds in the plug-in.

After successful extraction, you need to assign a <u>Category (on page 34)</u> to each sound and check their <u>Playing Ranges</u> (on page 35).

### Important:

Work like this should be done globally in **Audio/MIDI Setup**, rather than in an **Arrangement Rack**. This makes sure your extracted device will not be confined to a single arrangement.

# 7) Tip:

In case your plug-in does not support this function, you may want to request this feature from the developers. Usually it is relatively easy to implement. The more users are requesting this feature, the more likely the developers will follow your suggestion.

# Scanning Playing Ranges

If a sound is hosted by the **Audio Engine** or a **Drone**, Synfire can scan its <u>**Playing Ranges** (on page 35)</u> automatically by listening to its sound output.



Before you start, make sure a Category (on page 34) is assigned to the Sound (on page 37) or Instrument (on page 33).

Ö

Probe the plug-in for its pitch range estimate all three playing ranges. This may take while.

Option menu that allows you to copy and paste ranges, or distribute them equally or in proportion to known physical instruments.

# **Configuring Rack Modules Manually**

Using the <u>Sound Wizard (on page 193)</u>, or dropping items from the **Rack Module Library** are certainly the most convenient ways to add a new module to a rack. Still, you might find yourself in a situation where you need to configure a module manually from scratch.

# **Add Empty Module**

Double-click in the empty area of the rack to add an empty module based on the currently selected host. By default this is the preferred **Audio Engine**, unless some other host is selected on the tabs over the rack area.

If you want to use a **Drone**, **Drop** a yet unused one from the **Rack Module Library**, which will also add an empty module.

# Load Plug-in

Hover over the **Plug-in Button** and click on the double-arrow icon to select an audio plug-in to load. The editor of the plug-in will open, unless you are using a Drone. In the latter case you need to switch to the DAW and open the plug-in editor there. Synfire can't force the DAW to do that for you, unfortunately.

Configure the plug-in's sounds to your liking.

# **Select Device Description**

Unless the module is for an audio effect, it needs a **Device Description**, so Synfire can know what kind of sound the plug-in is providing. Pick one from the menu or create a new one as explained here (on page 234).

### Save a Preset

Once you have set up a rack module that is likely useful on other occasions, you should save it as a preset. This will make it appear in the **Rack Module Library** on the sidebar, from where you can conveniently drop it on future racks and instruments, so you won't have to configure it again.

# Synchronizing With a DAW

Be sure you understand the general implications (on page 283) of this.

### **Preparation**

- 1. Load as many **Drone** plug-ins into the DAW as you need. Don't mix VST, VST3 and AudioUnit Drones. Settle for one format and stick to it for the project. Guest plug-ins may use any format, though.
- 2. The <u>Sound Wizard (on page 193)</u> can now see the Drones. They also show up in the **Rack Module Library** from where you can drop them on instruments or into racks. Follow the wizard's instructions.

Alternatively, you can visit the **Sounds** tab and drop Drones from the **Rack Module Library** to the rack and load the desired plug-ins and device descriptions manually.

- 3. Switch to the DAW and configure the guest plug-ins of each Drone as needed.
- 4. Switch to Synfire and save your project.

#### Important:

Always save the Synfire arrangement first and then the DAW project. When opening, do it the other way around: Always open the DAW project first and wait 10 seconds until all drones are online. Only then do you open the corresponding Synfire arrangement.

### Synchronize Using MIDI

This works just as fine, although tempo changes can't be transmitted to the DAW.

- 1. Call Playback > External Synchronization and visit the MIDI tab on the Parameter Inspector (on page 66).
- 2. Check Enable External Sync and Enable MIDI Sync.
- 3. Make sure tempo in the DAW is the same as in Synfire.
- 4. Select a MIDI port for sending synchronization messages.
- 5. Select a protocol:
  - MIDI Clock: Transmits MIDI clock ticks and MIDI transport control messages (START, STOP, CONTINUE) and Song Position Pointers (SPP).
  - MIDI Time Code: Transmits SMPTE messages (full position pointers and quarter frame messages). Be sure to verify SMPTE offset and frame rate on the **Timing** tab.
  - Open Sound Control (OSC): This is still experimental.
- 6. Customize the protocol only if the default doesn't work. You may select different protocols for specific messages to ensure best compatibility with your external synchronization target.

### Note:

Every DAW might need different settings to work with Synfire. Please look up the required settings in the appendix <u>Synchronization Settings For DAWs (on page 330)</u>.

### Synchronize Using ReWire

If your DAW supports ReWire, synchronization is easy to set up. In order to start a new project immediately on a DAW instead of using an Engine first, you need to proceed as follows.

- 1. Load the ReWire module **Cognitone Transport** into the DAW. It does not generate audio signals. Make sure it is allowed to change tempo of the DAW.
- 2. Call Playback > External Synchronization and visit the Transport tab on the Parameter Inspector (on page 66).
- 3. Check the **ReWire** switch. If the name of your DAW transport shows up, both programs are synchronized.

### **Testing The Setup**

Verify that synchronization is working by selecting random containers or locations on the time ruler. The playhead of the DAW should follow your selection. Pressing start and stop should make the DAW follow suit.

### Important:

Remember that, depending on the synchronization method, you may need to set the tempo in the DAW manually in order to match that of Synfire. This is extremely important for proper synchronization.

### **Related information**

Synchronization Settings For DAWs (on page 330)

Audio Engine vs. DAW (on page 283)

# **Relocating From Engine to DAW**

### Situation

You have a self-contained arrangement with all sounds based on the **Audio Engine**. You want to continue composing in a DAW using the same sounds and synchronize both programs. The task is to move the sounds over to the DAW (where they are hosted by <u>Drones (on page 43)</u>) and make it a synchronized Synfire + DAW project from now on.

### How to Relocate

Step-by-step instructions how to accomplish the task.

- 1. Create a new project in your DAW using the same name as the arrangement.
- 2. Synchronize your DAW with Synfire as explained here (on page 238).
- 3. For each module on the <u>Arrangement Rack (on page 24)</u>, load a <u>Drone (on page 43)</u> plug-in into the DAW. Don't mix VST, VST3 and AudioUnit Drones. Settle for one format and stick to it for the project. Guest plug-ins may use any format, though.
- 4. Visit the **Sounds** page and for each rack module pick the desired target Drone from the port menu. Synfire will move the plug-in and its current contents over to the Drone and update all affected instruments accordingly.
- 5. Repeat the last step until all plug-ins have moved to Drones in the DAW.

Save your Synfire project first, then save the new DAW project. Preferably keep both files in the same folder.
 Now you have a synchronized Synfire + DAW project.

# i Tip:

If you can not hear the Drones playing, you may need to check if the DAW requires tracks to be armed for monitoring in order to render sound output while the transport is idle.

i Tip:

You can also relocate a project from a DAW back to an Audio Engine. Or from one Audio Engine to another, provided the same plug-ins are available on both computers.

i) Tip:

Always move all plug-ins of an arrangement to the same host, or you will experience timing problems.

#### **Related information**

Drones (on page 43)

External Synchronization (on page 66)

# Inspecting The Audio Engine

Besides visual feedback on audio output and checking which otherwise invisible plug-ins are currently loaded, opening the Engine user interface is of little use. Still, in the event you need to, here's how to do it.

- 1. **Windows:** Find the **Engine** icon on the Windows task bar and double-click it. This will open the user interface of the engine.
- 2. Mac: Find the Engine icon on the dock and bring it to front. You can the access all functions through the main menu bar.

# Editing

Common tasks while you are developing a project

# **Playing Palettes**

How to navigate and make sense of a palette

#### Video Tutorial

Simply click on a chord in a **Palette** to schedule it for live playback. Drag any chord to a progression to insert it. Better yet, review, edit and drag off the <u>Harmonic Context (on page 188)</u> panel of the right sidebar.

There are two playback modes you can choose from.

### **Chords-Only Mode**

- Enable Playback > Chords-Only Mode to hear chords played with the Global Instrument 'Chords'.
- Disable *Playback > Sustain Palette Chords* if you want the chord to stop once you release the mouse button.
  Otherwise it will be sustained until you schedule the next.
- Enable Playback > Play Bass Instrument to hear the bass played with the Global Instrument 'Bass'.

### **Sketch Mode**

Select a Sketch from the menu Playback > Sketches to render all chords with a real-time accompaniment.

# Note:

Always click on the next chord a short moment ahead of time, so Synfire gets a chance to prepare for the change. A few milliseconds are sufficient.

### **Transposition, Inversion And Voicing**

In **Chords-Only Mode** you can control inversion and voicing of a chord to some extent by using the keyboard commands mapped to the *Transform* menu (open the menu to see).

Once a chord was added to a progression, there are two ways to transpose its sound.

- If an entire progression seems to be pitched too low or high, you should transpose or widen the middle Playing Range of the Global Instrument 'Chords'. Open Playback > Audio/MIDI Setup and edit the middle playing range of the 'Chords' instrument.
- If individual chords sound pitched wrong, you can edit Inversion, Voicing and Alignment in the Progression Editor or on the <u>Harmony Lane (on page 97)</u>. If you need to remove these edits from a progression again, do Transform > Reset Hints.

### Important:

Remember that it is up to the individual **Instrument** and the <u>Interpretation (on page 301)</u> parameter whether to obey the hints regarding inversion, voicing and alignment in a progression. The built-in interpretation for Chords-Only playback does so, but those your arrangements may not.

# **Trying Alternative Chords**

While playing a palette you can use the Left or Right arrow keys to return to the previous chord or advance to the chord previously returned from (back and forth). Then you can try an alternative path starting from there.

The <u>Harmonic Context (on page 188)</u> panel on the right sidebar shows a back-trace of the three most recently played chords.

### **Highlighting Relationships**

#### Chords

Select a chord to schedule it for playback. This highlights all scale steps used by the chord.

Drag a chord to any progression to insert it there.

#### **Horizontal Scales**

Select one or more scales to highlight chords that can be built from them exclusively.

#### **Scale Steps**

Select or more steps to highlight chords that include them.

#### **Advanced Navigation**

Using the *Navigation* menu, you can highlight in all open palettes the chords they have in common (**Shared**) or those they own exclusively (**Distinct**). This is helpful for finding a path to a key change (on page 10) (modulation).

#### Harmony (Outlet)

Drag the Notepad progression off to any destination, or drop a progression from elsewhere.

#### Freeze

Locks current chord and scale selections while you continue playing.

#### **MIDI Input**

Highlight scale steps and chords according to MIDI input.

#### **MIDI Live**

While responding to live MIDI input, also detect incoming chords.

#### **Surfing Multiple Palettes**

Open multiple palettes and play chords on them in any order. This is helpful if you want to compose a <u>key change (on</u> <u>page 10)</u> (modulation). Everything you play in one palette will be visualized on the other as seen from the perspective of that key.

#### **Retrospective Recording**

It is possible to call up the last played chord sequence at any time with *Navigation > History* .... This opens a **Progression Editor** where you can copy & paste the recorded chords.

#### **Related information**

Editing Palettes (on page 108)

# **Moving Parameters**

Copy or move parameters to other instruments, containers, libraries or windows

<u>Parameters (on page 2)</u> can be moved around freely for unlimited experimentation. When a parameter is copied to a destination with a different type, its <u>data is converted automatically (on page 324)</u> to match.

#### **Drag & Drop**

- 1. Click on a lit Parameter Outlet (on page 45).
- 2. Hold the mouse button and drag it towards the desired destination, which can be another outlet, an instrument, a container, a library or a different window.
- 3. Hold the  $\sim$  Alt key if you want to make a copy, otherwise it will be moved.
- 4. Drop it over the destination by releasing the mouse button.

#### **Copy & Paste**

Alternatively you can use the keyboard.

- 1. Click on a lit Parameter Outlet (on page 45).
- 2. Press #C Control-C
- 3. Click on a destination outlet.
- 4. Press XV Control-V to paste the data.
- 5. Or use the Right-Click (on page 44) menu to copy and paste.

#### **Related information**

Parameter Outlet (on page 45) Parameter Conversion (on page 324)

# **Editing Parameters**

In the Parameter View you can edit parameter data in detail. Where parameter data is shown in gray, it is inherited or shows for additional background information. If you want to edit inherited data, make a **Snapshot**, or edit it in the original container.

#### **Manipulating Selections**

All commands you call from a menu or the keyboard are applied to the entire parameter, unless a **Selection** is established. Use the tools listed below to make a selection. Click on a selection again to undo it.

£

Span Selection: Extend a selection by holding down 🕆 Shift and clicking at the target position.

Drag at the edges to stretch or compress its content in any direction. If you want content to the right of a selection to make room while you stretch it, or close the resulting gap while you shrink it, hold down **E** Control.
Move or transpose the selection with the Arrow Keys. The current **Grid** applies. The Left and Right arrow keys scroll the selected content around. If only a single position is selected (vertical bar), the object will move along with it.

## []

Chord Selection: Select and manipulate chords in a Harmony (on page 300) parameter.

## 8

Pointer Selection: With the arrow keys, you can move and transpose it.

Drag a lasso rectangle to select multiple objects, or extend a selection by holding down **#** Control while you click on additional objects.

Transpose or move the selection with the Arrow Keys . Hold down 🕆 Shift for finer steps.

### ഹ്ല

Symbol Selection: Move, transpose, stretch and delete the selection.

Drag a lasso rectangle to select multiple symbols, or extend a selection by holding down **# Control** while you click on additional symbols.

Transpose or move the selection with the Arrow Keys . Hold down 🛈 Shift for finer steps.

#### **Edit Menu**

Several commands can be called from the *Edit* menu when the *Parameter View* (on page 144) has input focus.

#### Cut, Copy, Paste, Duplicate, Clear

Does as the name implies with the current selection.

#### **Paste Rhythm**

Transfer the (implied) <u>Step (on page 310)</u> of the object currently on the clipboard to the selection or entire parameter. This repositions selected objects in sequential order. Unless an actual <u>Step (on page 310)</u> parameter is on the clipboard, the steps inserted here are derived from the positions of objects in a parameter.

#### **Paste And Merge**

Paste additional data (at the selected position) without overwriting existing data. Works for polyphonic parameters only.

#### Group, Ungroup

Makes a new Figure (on page 294) segment from all selected symbols or segments, or separates all symbols of the selected segment respectively.

#### Divide

Splits a Figure (on page 294) segment at the selected symbol.

## i Tip:

Transfer the rhythm of one segment to another: Select a segment or a span on the time axis and copy it. Then *Edit > Paste Rhythm* to apply.

## i Tip:

Change the rhythm of a chord progression: Copy a span of the progression that represents the desired rhythm and *Edit > Paste Rhythm* to apply it to the entire progression.

#### **Parameter Menu**

#### **Insert Snapshot Of Parameter**

Inserts a physical copy of an inherited parameter so you can modify it.

#### Insert Template ...

Pick preset parameter data from a menu.

#### Save Template ...

Save current parameter data as a template for later reuse.

#### **Extract From MIDI Output**

Looks at the actual MIDI output and extracts the current parameter so you can edit it.

#### **Save To Figure**

Applies the parameter to the Figure and clears it after that.

#### **Extract From Figure**

Looks at the Figure (on page 294) and extracts the current parameter so you can edit it.

#### Repeat

Parameters are looped by default, that is they repeat until the end of the container. If you disable this with *Parameter > Repeat*, the parameter won't supply any more values beyond its length. For some fundamental parameters like Figure (on page 294), this means no more notes will be generated.

#### Interpolation

With *Parameter > Interpolation* you can determine which values a parameter delivers at positions where it has no data.

- 1. Recent: Deliver the most recent value until the next value occurs.
- 2. Snap: Deliver the value that is nearest to the position being asked for.
- 3. Linear: Interpolated continuously from one value to the next.
- 4. Dithered: Increase the random probability of delivering the next value the closer we get to it.

- 5. **Sequential**: Deliver the next value every time. Position is completely ignored. Once all values are delivered, continue from the beginning. Allows for interesting effects with algorithmic music.
- 6. **Hits**: Deliver a value only at its exact position. Anywhere in between the value is undefined (same as if the parameter is not looped and has reached its end).

## Note:

This feature is available with the Express and Pro editions.

## **Transform Menu**

Which commands are available on the *Transform* menu depends on the type of parameter currently selected. There are <u>additional commands (on page 249)</u> specifically for the Figure parameter and <u>special commands (on page 252)</u> for the Harmony parameter.

#### **Dynamics**

Emphasize light or heavy beats by different amounts.

#### Reverse

Reverse the selection as if it were played backwards ("tape reverse").

#### **Reverse Preserving Rhythm**

Reverse the sequence of values while retaining steps and lengths. For example, you may duplicate a melody and reverse it this way to have both melodies play opposite pitches following the same rhythm.

#### Flip

Flip the selection upside down.

#### **Invert Up/Down**

Inverts chord segments upwards or downwards.

#### **Stretch Parameter**

Lengthen or shorten the entire parameter by a set amount.

#### Quantize

Snap positions and lengths of objects to the current Grid.

#### **Clip Lengths**

Snap lengths of objects to the length suggested by the current **Grid**, or keep them if they are already shorter.

#### Resample

Scan the current selection using the current **Grid** and rebuild it with interpolated values computed at those positions.

#### Monophonic

Reduce a parameter to a only single value at any time.

#### **Drop Repetitions**

Drop repeated occurrences of the same value.

#### **Drop Redundancy**

Drop values that are unnecessary in light of the current Interpolation.

## **Resizing a Parameter**

Drag at the resizing handle > to alter the length of a parameter. If you shorten a parameter, data beyond its end will not be deleted, although you may no longer see it.

### **Related information**

Editing Figure (on page 247) Editing Harmony (Progressions) (on page 250)

## **Editing Figure**

**Figure** is a powerful parameter with unique editing options. This section is about the things that are different with this parameter. For general parameter editing, please refer to the section <u>Phrase Editor (on page 86)</u>.

### Selection

## $\mathbf{k}$

Segment Selection (Pointer): Click on a symbol to select the segment it belongs to.

- Click on the selection again to switch to the Symbol Selection tool.
- Transpose or move the selection with the Arrow Keys . Hold down 🕆 Shift for finer steps.
- Drag a selection to move it. Hold down  $\sim$  Ctrl to make a copy.
- Drag at the top, bottom or right end to stretch the segment.
- Drag the right end while holding  $\sim$  Alt to alter the length of all symbols.
- Hold down **# Control** and drag up or down to alter the velocity of all symbols proportionally by a percentage.
- Hold down **#** Control to unlock the grid while you are resizing or moving an object.
- Ungroup a segment with **#U** Control-U.

## ഷ്

Symbol Selection: Click on a symbol to select it.

- Click on the selection again to switch to the Segment Selection (Pointer) tool.
- $\cdot$  Drag a selection to move it. Hold down  $\sim$  Ctrl to make a copy.
- Transpose or move the selection with the Arrow Keys . Hold down 🕆 Shift for finer steps.
- Drag the right end to alter the length of the selected symbols.
- Hold down **#** Control and drag up or down to alter the velocity of selected symbols absolutely.

- Hold down 😹 Control to unlock the grid while you are resizing or moving an object.
- Make a symbol become the anchor of its segment with **#R** Control-R.

#### **Multiple Selections**

To select multiple symbols or segments, hold down **# Control** while you click on them, or drag a lasso rectangle around them.

Select all segments in the Figure (on page 294) with #A Control-A

Group multiple symbols or segments to form a new segment with #G Control-G.

#### **Span Selections**

Keep in mind you can use the **Span** tool with <u>Figure (on page 294)</u> too. Use it to scroll the positions of segments around or transpose an entire <u>Figure (on page 294)</u> with the Arrow Keys.

### Drawing

Use the Line and Freehand tools to draw new symbols and segments.

- · Click on a target position to insert a single Symbol.
- Draw a line or curve to insert new symbols at positions that match the current **Grid**. The **Anchor** of a new segment is placed where you start the stroke. Thus, stroking backwards will put the anchor at the right end.
- Draw a vertical line to insert a Chord.
- Draw a horizontal line to create a single **Symbol** with a particular length. Enable <u>Duplicates (on page</u>) if you want multiple equal symbols be inserted along the current **Grid**.
- Draw a diagonal line to insert symbols where it crosses the current **Grid**. Enable <u>**Duplicates** (on page</u>) if you are fine with symbols being repeated.
- To append additional symbols to a segment, hold down **Shift** in the proximity of that segment and start drawing.
- Try the Scale Board to assist you with drawing segments incrementally.

## Rhythm

You will notice that drawing a random line or freehand curve on the Figure (on page 294) parameter looks instantly pretty but doesn't always work musically. The reason is rhythm. This is what you can do to ensure consistent rhythm.

- Be sure to select a <u>Grid (on page 88)</u> that makes sense for the kind of phrase you have in mind. You can drop another phrase on the <u>Grid Outlet (on page 88)</u> to exactly recreate its rhythm.
- Start and end the line or curve at rhythmically significant positions. Consider their weight (heavy or light).
- Look at other phrases that work and resemble their structure.

#### **Dynamics**

Apart from very basic lines and sparsely placed notes, a Figure (*on page 294*) without expressive <u>Velocity (*on page 317*</u>) is bland. This is what you can do to add dynamics.

- Open the Click Board and use it to emphasize heavy or light parts of the beat.
- Recall a template from the right-click menu over the Velocity (on page 317) outlet.
- Switch to Hyper Edit mode and use the Shape tool to sketch dynamics.
- Select a segment and hold # Control while you move the mouse up or down to change dynamics proportionally. Hold down 
   Shift in addition if you want to change values absolutely.
- Select <u>Velocity (on page 317</u>), click into the Parameter View and use the arrow keys to transpose values up or down.
- Transfer dynamics of another phrase by dropping <u>Velocity (on page 317)</u> or <u>Figure (on page 294)</u> of that phrase on the <u>Velocity (on page 317)</u> outlet of the target phrase.

# 7 Tip:

To save the clutter of an extra <u>Velocity (on page 317)</u> parameter, you can edit dynamics of a phrase in **Hyper** Edit mode. Alternatively, edit a physical <u>Velocity (on page 317)</u> and permanently apply it to the figure with Parameter > Save To Figure when done.

## **Hyper Edit**

After switching to this mode with the tool **H**, you can edit the <u>Velocity (on page 317)</u>, <u>Step (on page 310)</u> and Length of a <u>Figure (on page 294)</u> as if they were physically present parameters. In reality however, your changes are immediately saved to the figure.

## Note:

**Hyper Edit** mode makes sense only where a **Figure** (on page 294) is present. If necessary, make a **Snapshot** beforehand.

## **Transform Menu**

The menu Transform offers additional commands for the Figure parameter.

#### **Reverse at Anchor**

Flip segments horizontally at their anchor positions. In contrast to other reverse commands, this creates a wildly different figure.

#### **Reverse at Onsets**

Play segments backwards without altering symbol lengths. In contrast to the default "tape reverse", this turns a legato melody into partially overlapping notes.

#### **Flip at Anchor**

Mirror segments vertically at their anchor position.

#### **Force Legato**

Alter the lengths of symbols such that overlaps are removed and all gaps between them are closed.

#### **Shorten Overlaps**

Shorten symbols only where they overlap in time.

#### **Remove Clashes**

Shorten symbols only where two repeated symbols overlap.

#### **Simulate Strumming**

Turn a vertical chord into a strummed performance.

#### **Simulate Pedal**

Stretch all symbols to play until the end of the last symbol in a segment.

#### **Simulate Broken Chord**

Turn a vertical chord into a broken, arpeggio-once performance.

#### Select

Select all segments of the current type.

#### Focus On

Highlight segments of the current type and dim all others.

#### Change

Convert all selected segments to the current symbol type.

#### **Simplify Chords**

Reduces chord segments to fewer symbols and adjusts segment properties to render the chord as it is configured with the Harmony parameter.

#### **Round Accidentals**

Round all symbols to diatonic steps.

#### **Remove Symbol Tags**

Drop all tags from a take (Take only).

#### **Related information**

Collecting Phrases (on page 281)

Optimizing Phrases (on page 254)

# Editing Harmony (Progressions)

#### Video Tutorial

**Harmony** is a powerful parameter with unique editing options. This section is about the things that are different with this parameter. For general parameter editing, please refer to the section <u>Phrase Editor (on page 86)</u>.

# []

#### Chord Selection:

- Click to select a chord.
- Click on another chord while holding down 
   Shift to extend a selection.
- Click into the gap between two chords to paste something in between them.
- Drag at the right edge of a chord to change its length. Hold down **#** Control if you want to make room while stretching or close the gap while shrinking.
- Use Arrow Keys to move or extend a selection like you would do in a word processor.
- Transpose chords with Up/Down Arrow Keys while holding down H Control

# £

Span Selection: Select and edit arbitrary spans independently.

#### Drag

Drag a selection to move it. Hold down <u>Alt</u> to copy. You can drop it also on any <u>Harmony (on page</u> 300) outlet, <u>Container (on page 21)</u> or Phrase.

#### Drop

Drop a chord from a Palette to insert.

Drop a (partial) Progression (on page 14) from elsewhere to replace, or hold down **# Control** to insert.

## Colors

Each context shows a background color that hints at the **Relation Key**, using the colors also seen on the **Circle Of Fifths**. With a little getting used to, this can help you see at a glance what's going on in a progression, especially where the scale material changes more dramatically.

## i Tip:

To clean up a progression with excessively many different colors (which is not a good sign in most cases), do a *Transform > Estimate Key, Relations and Scales*. Using the **Alternative** option for **Scale Selection Preferences** often smoothes out the edges very well.

## Pace & Rhythm

The pace and rhythm of chord changes is an important factor. A change at every measure can be boring. A change at every quarter note is often too fast to be noticeable and will muddy the overall experience. A good place to start is a change every two quarter notes, or roughly in the middle of a measure. It keeps harmony moving and is slow enough for instruments to pick up enough notes to make a chord noticeable.

## i) Tip:

Try a progression at half or double speed (by stretching it) to find the pace that best matches your song. Also try different durations for each chord.

## i Tip:

You need not nudge the onset of a chord change to the left in order to meet an instrument's rhythm. Instead, adjust the **Look Ahead** setting of the Interpretation parameter.

#### **Inspector Tabs**

#### Harmony

Edit all components of the selected Harmonic Context (hover over widgets for help).

#### Scheme

The metric scheme edited here is saved to the <u>Harmony (on page 300)</u> parameter of the current container. The length of the progression is automatically calculated based on your input.

A certain column layout results from applying the metric scheme, which breaks the progression down into rows. Use the *Transform > Column* menu to split or cleanup the columns where needed.

#### Layer

Use the *Transform > Layer* menu to create or remove layers that you can select on the **Layer** tabs at the top and edit here.

#### **Scale Selection Preferences**

These preferences are used when you do *Transform > Estimate Key, Relations and Scales* or by pressing the Estimate button.

## **Details Shown**

On the *View* menu, you can hide or show individual components of a harmonic context to show up in the progression or not. There is only limited space, so not all components can be shown at the same time.

#### **Transform Menu**

#### Estimate Key, Relations and Scales ...

Apply key detection and scale selection to the current selection according to current **Scale Selection Preferences**.

#### **Estimate Scales Only**

Apply only automatic scale selection according to current Scale Selection Preferences.

#### Replace Key ...

Set a specific key for all selected contexts.

#### **Replace Relation ...**

Set a specific relation for all selected contexts.

#### **Automatic Bass**

Reset bass to automatic selection.

#### Bass: Prime (Root), 2nd, ... 9th

Demand a specific bass interval.

#### **Toggle Alignment**

Enable or disable pitch alignment between chords.

#### **Reset Hints**

Remove all hints regarding inversion, voicing and alignment.

#### Jazz Up / Down

Add or remove chord extensions.

#### **Transpose**

Transpose current selection to absolute pitches or by a relative amount.

#### Column

Split or cleanup the column grid of the progression according to the metric scheme.

#### Layer

Add or remove polytonal layers.

#### Reverse

Reverses all chords as if a tape was played in opposite direction.

#### **Reverse Values**

Reverses the ordering of chords, but keeps transition positions unchanged.

#### Flip

Experimental inversion of harmony.

#### **Drop Repetitions**

Removes repeated contexts, provided all their components are equal.

#### **Open Suitable Palette ...**

Open a Palette on the current key (or: Scale Set).

#### **Sections**

The standalone **Progression Editor** window allows for multiple progressions to be saved under different section names. Use the *Section* menu to add, select, rename or remove sections of a file.

i Tip:

Don't use the sections for collecting progressions. A Library is a much better tool for the task.

## **Controlling Inversion, Voicing and Octave**

Harmony provides the rules for all instruments. Each instrument may interpret them in different ways. What you hear in the progression editor is how the global instrument **Chords** interprets them. In order to control inversion, voicing and pitch range for a phrase in your arrangement, you need to edit its **Figure** and **Interpretation** parameters.

- 1. Draw chords with **Chord** type symbols by dragging a vertical line with the **Line** tool.
- 2. Transpose symbols up or down to control inversion.
- Edit the properties of a Figure segment on the Parameter Inspector to control voicing, bass and alignment with the previous chord.
- 4. Edit the middle Playing Range of the instrument to control the general pitch range, e.g. the octave.

## **Combining Multiple Progressions**

Unlike audio or static MIDI clips, the Harmony parameter can be easily transposed in order to work in every key. Just transpose a progression (or a part of it) up or down until the global key setting matches the key you need. Once they are transposed into key, partial progressions can be combined freely.

#### **Related information**

Collecting Harmony (on page 282)

# **Optimizing Phrases**

Tweak phrases to make them work best in any context

A comprehensive online tutorial can be found here.

## Harmonizing a Phrase

## How to turn one or more takes into a Harmony (on page 300) parameter

Multiple views of Synfire feature a **Harmonizer** that turns one or more **Takes** into a <u>Harmony (on page 300)</u> parameter guided by your instructions and feedback. The final result is saved to the current <u>Container (on page 21)</u> or **Phrase**.

This way you find chords that match a MIDI recording or the rendered output of one or more instruments.

To open a Harmonizer, select one or more instruments on the **Track Sheet** and do *Phrase > Harmonize*. This takes you to the **Harmonizer** tab with the selected instruments ready for harmonization. You can also just switch to the tab at any time.

## i Tip:

To harmonize a melody you have in Standard MIDI File format, first select the desired instrument and import the MIDI track with *Phrase > Import* with the figure recognition preferences set to **Input for Harmonizer** (static).

#### **Workflow Summary**

- 1. Select one or more instruments as input.
- 2. Verify the global key and other settings.
- 3. Generate chord suggestions with Harmonize
- 4. Listen to chord suggestions and make a choice.
- 5. Add or remove chord transitions as desired.
- 6. Accept the final result with Apply

#### 1. Select Inputs

Select one or more instruments as input source. The <u>Take (on page 312)</u> parameter is used by default. Alternatively you can use rendered <u>Output (on page 305)</u> for interesting surprises. The **Parameter View** shows a piano roll style representation of the input.

#### Troubleshooting:

If you recorded a melody incrementally in multiple steps, the <u>Take (on page 312)</u> might hold only the most recently recorded input. Make sure you have a complete take before harmonizing it.

# 7 Tip:

Use static pitch symbols for vocal melodies, if you want to arrange a song around them. These won't change when they are rendered against new harmony.

### 2. Global Key And Settings

#### Key

Click on this button or call *Transform > Pick Global Key* to select a global key from the menu. If a <u>Harmony (on page 300)</u> parameter already exists, its key is considered with priority. This setting has a profound influence on the resulting chord suggestions.

#### **Transitions**

Algorithm that determines the positions of chord changes.

- 1. Auto: Selects an appropriate algorithm automatically.
- 2. Search: Looks at the input and estimates where chord changes can most likely be assumed.

- 3. **Polyphony**: Assumes every cluster of notes is a new chord. This algorithm is very simple and works with sequences that contain sustained chords only.
- 4. **Keep**: Keeps existing transitions unchanged. In order to protect your edits, this mode is enabled automatically after you changed a transition manually.

#### Style

Influences the complexity of chords and potential key changes being suggested. You should experiment to find the settings that best match your input.

#### Resolution

Clusters of notes whose onsets fall within this span of time are considered as potential chords. The setting helps with input that is not quantized.

#### **Shortest**

Potential chord changes faster that this will not be considered.

## 3. Generate Chord Suggestions

Click Harmonize to run the algorithm on the selected input. Selecting a span in the Parameter View (on page 144) will harmonize only the selected range.

Check the option **Retain Harmony** to use the current <u>Harmony (on page 300)</u> parameter to make initial default selections in the list of chords.

## CAUTION:

If you want to (re)harmonize the entire input as a whole, you must make sure you have nothing selected in the **Parameter View**. This can be easily overlooked.

# Note:

The **Generic** and **Simplified** styles deliver basic (vanilla) chords only. If you want more elaborate extensions like Am9 or c7(9,13) instead of Am and c7 respectively, you can add these extensions in the **Progression Editor** later.

## 4. Select Chords

Select any of the detected transitions and pick a chord from the list to use it for this transition. Although chords are sorted by probability, some chords further down in the list can also offer interesting solutions.

If only a single chord is listed, the selected transition was not yet harmonized. Press **Harmonize** to search for potential chords.

When you select a transition, the Global Instruments play a short preview to illustrates the result so you can assess it.

## i Tip:

There are thousands of possible chord progressions that match a melody. Harmonization is a creative decision. Trust your ears and personal style preferences to find the best results.

#### 5. Insert or Remove Transitions

Select a position or span in the **Parameter View** (on page 144) an press + or - to add and remove chord changes, respectively.

#### 6. Accept The Result

## $\triangleright$

Test your current choices in context of the arrangement.

## 60

Render a preview of the chords only.

#### **Match Neighborhood**

Synfire picks selections from the suggested list of chords automatically in order to optimize the overall progression. This affects transitions that follow the position you have changed. Hence, a large portion of the progression may change after a single edit.

#### **Optimize Scales**

Before the result is finally accepted for the container, all scale selections of the progression will be optimized.

#### Apply

Saves your current choices in the <u>Harmony (on page 300)</u> parameter. You may select a partial span to save a limited range only.

# 7 Tip:

The Harmonizer assist you with finding a chord progression, but it won't create a contrapuntal movement for you. If that is your goal, you need to create a melodic Figure for the desired instruments instead.

#### **Drag & Drop**

Drop a chord from a palette or progression on the **Parameter View** (on page 144) to insert it as a choice. Copy & paste works as well. No checks are performed. Whether it makes sense or not is your decision.

#### **Preview Mixer**

Use these faders to adjust the volume of the Global Instruments used to render results.

# Snapshots

## How to make a physical copy of an inherited or default parameter

Parameter data set in one container is also available in its child containers. Where parameter data is set, the <u>Parameter Outlet (on page 45)</u> is fully lit (bright), while in sub containers the outlet is only partially lit (dimmed). If you want to modify inherited parameter data, you need to make a **Snapshot** of it first.

- 1. Select the desired parameter outlet.
- 2. Right-click on it and choose Insert Snapshot Of Parameter from the menu, or press **CRS** Control-Alt-S
- 3. Alternatively you can use the button on the nearby toolbar.
- 4. A (partial) copy of the inherited data is inserted.

## 7 Tip:

If you want to modify only a small portion of inherited parameter data, you can insert a sub container spanning the desired range and make a snapshot of the parameter in there. This will keep the rest of the parameter untouched.

## **Container Snapshots**

*Container > Make Snapshot* collapses and removes all sub containers and makes snapshots of all inherited parameters for all instruments. Not unlike bounce-to-disk in a DAW, it is a way to consolidate nested containers into something more simple and lean.

Be aware though, that some parameters like <u>Interpretation (on page 301)</u> or <u>Scheme (on page 308)</u> are constants within a container. Their value can't change. Any changes with these parameters in sub containers will be lost after a snapshot.

#### **Related information**

Containers (on page 21)

# Fine Tuning Interpretation

## How to allow more dissonance and free wheeling harmony

There are several ways to tweak and influence voice leading process.

## **Allowing More Dissonance**

The **Voice Leading** process (VL) ensures that melodic movements are preserved no matter what happens to <u>Harmony</u> (on page 300) or how narrow the current **Playing Range** may be. Depending on the strategy you selected, it also takes care that chord changes are emphasized by melody where possible.

While transposing figure symbols in the phrase editor, you might notice that certain pitches are avoided. This is because VL excludes them for various reasons. If you feel a specific pitch should absolutely be allowed, you can do one of the following to relax VL constraints.

#### **Choose a Different Scale**

Your Vertical Scale may be too tame. Try a more dissonant one.

#### **Insert an Intermediate Chord**

If you have a specific melody in your head for which VL is not allowing individual pitches to be placed, this is probably because the chord in your head is different from the one in the progression at that point in time. Try adding intermediate chords or doubling the pace of your progression. Fast chord changes are not necessarily perceived as such, while they extend the scale material allowed for melodies.

#### **Add Chord Extensions**

Since the **Vertical Scale** is responsible for the set of allowed pitches, adding extensions to a chord makes sure these notes are more likely to be allowed. Still, some strategies allow only the strongest chord intervals to be used at rhythmically important places, excluding some chord extensions.

#### **Lower Strength**

Override the strategy by lowering the Strength (on page ) parameter.

#### **Allow Minor Seconds**

Check **Enable Dissonance** on the **Form** tab. This will allow the minor seconds in some chord inversions to be played. This however only affects **Chord** segments.

#### **Disable Cooperation**

Disable the Cooperative mode.

#### Select a Different Strategy

Select a more relaxed VL strategy from the pop-up menu.

#### **Disable Voice Leading**

Disable VL for the individual Figure (on page 294) segment.

#### **Allow Chromatic Alterations**

In addition to disabling VL, also allow chromatic alterations for the individual Figure (on page 294) segment. This is really only a last resort, as it makes your phrase somewhat hard-wired to the current Harmony (on page 300). It won't adapt well anymore when you move it to another place.

## Fitting Music Between Cue Markers

#### Using the Time parameter to fit music between cue markers.

How to adjust the tempo of a musical passage automatically in order to have it fit between two events in a film.

- 1. Select a cue marker in the Time (on page 314) parameter.
- 2. Select a position on the time ruler.
- 3. Press the button Match with Ruler. Synfire changes <u>Tempo (on page 313)</u> such that your selected position on the time ruler exactly matches the absolute time denoted by the cue marker.

#### Note:

This method is only useful within certain limits, because Synfire assumes a tempo outside the range of 30 to 300 does not make sense. Instead of extreme values for <u>Tempo (on page 313)</u>, you should insert or delete a number of measures (bars) in order to get closer to the desired duration before doing this match.

### **Finding Ideal Tempo**

By selecting both parameters <u>Tempo (on page 313)</u> and <u>Time (on page 314)</u> at the same time, you can watch the cue markers move relatively to the music while you change the value for Tempo.

This way you can adjust the tempo such that events in the film roughly meet with corresponding positions in the music, before you begin using the matching method explained above.



This feature is available with the Pro edition.

# Recording

How to record phrases and parameters and live-detect incoming chords

#### Video Tutorial

Converts captured MIDI input into a **Phrase** or individual **Parameters**. MIDI input is captured in the <u>Take (on page 312)</u> parameter and subsequently **Figure Recognition** is run to create a **Figure** (on page 294).

#### Workflow

- 1. Select <u>Container (on page 21)</u> and Instrument of the phrase to record into.
- 2. Select a starting position or span, if the recording shall be limited to a particular area (more on this below).
- 3. Press 🔾 and perform whatever you want to record on your external MIDI keyboard.
- 4. Press . Synfire converts the <u>Take (on page 312)</u> into a <u>Figure (on page 294)</u>, based on the last used settings for figure recognition.
- 5. Press Figure Recognition Settings to return to the <u>Take (on page 312</u>), if you want to try the process with different settings until you are satisfied with the result.

#### **Selective Recording**

You can limit a recording to a particular span inside the phrase. If prior to recording a starting position is selected with zero duration (vertical line only), the recording will replace everything thereafter. If a span with some duration is selected, content outside the selection is protected from being overwritten.

This way you can work your way through a longer phrase incrementally, one short take at a time.

### Note:

Keep in mind that only the most recent recording is kept in the <u>Take (on page 312)</u> parameter. If you record multiple times, previous takes will be replaced.

#### Latency

Synfire compensates for MIDI latencies if you enable **Compensate Latency During Recording** in *Playback > Audio/MIDI* Setup > Inputs and use the Audio Engine for MIDI input.

Latency compensation is based on timestamps in the MIDI messages being received. Only the Audio Engine is capable of passing MIDI timestamps to Synfire.

#### **Related information**

Recording Parameters (on page 261) Setting Up MIDI Input (on page 233) MIDI Routing (on page 41)

## **Recording Parameters**

Instead of recording a Figure (on page 294), you can also record many other parameters. For example, you may want to record MIDI controllers, Velocity (on page 317), Step (on page 310) or Rhythm (on page 307) by tapping them on your external MIDI keyboard. The record button is grayed out if the currently selected parameter doesn't support recording.

- 1. Select the desired parameter.
- 2. Press 🗠 on the transport bar and tap your performance on the external MIDI keyboard or move a controller.
- 3. Check the result in the Parameter View (on page 144) of the phrase. You may need to make Parameter Views

(on page 144) follow parameter selection in order to see it: Click 🔀 above the Parameter Block (on page 143).

## Note:

Parameter recording also works in **Hyper Edit** mode, so you can alter the rhythm and dynamics of a phrase by recording <u>Velocity (on page 317)</u>, <u>Step (on page 310)</u>, <u>Length</u> respectively.

#### **Recording Harmony**

Although Synfire can detect chords from live MIDI input, it is fundamentally difficult to determine proper chord names without context. Thus, you will get much better results if you simply record what you are playing live on a **Palette**.

- 1. Open one or more **Palettes**.
- 2. Select a suitable **Sketch** or choose **Chords-Only** mode.
- 3. Select the parameter Harmony (on page 300).

- 4. Press the parameter record button on the transport bar.
- 5. Play chords on the palette or multiple palettes. Synfire adds all chords with full harmonic context to the progression until you stop the recording.

You can start the recording anywhere you choose (<u>Selective Recording (on page 260</u>)). You could even record in multiple windows at the same time.

### **Related information**

Setting Up MIDI Input (on page 233) MIDI Routing (on page 41)

# Live Chord Detection

Synfire can detect chords from live MIDI input. It makes Synfire double as an intelligent accompaniment machine. The detected harmonic context shows up in <u>Palettes (on page 108)</u>, <u>Keyboard Widgets (on page 185)</u> and the <u>Circle of</u> <u>Fifths (on page 183)</u>.

## **Enable Live Detection**

- 1. Make sure Synfire is receiving input from your MIDI keyboard.
- 2. Go to the Progression or Palette tab.
- 3. Make the Keyboard Widget (on page 185) visible.
- 4. Enable <u>MIDI Input (on page 233)</u>.
- 5. Enable Live Chord Detection
- 6. Play chords on your external MIDI keyboard.

## **How It Works**

A new chord is detected based on the keys you hold on the keyboard. The current key setting helps narrow down the search to plausible candidates.

- 1. At least three (3) notes pressed at the same time trigger a chord detection.
- 2. The lowest note is considered bass. It helps to play it at least an octave below the other notes.
- 3. Members of the current key and palette are preferred over more remote candidates.
- 4. Detected chords not yet visible in the palette will be added.

## Note:

If a global <u>Keyboard Split (on page 161)</u> is set, only keys below the split are considered for chord detection. Keys above the split are reserved for playing an accompanying melody.

*i* Tip:

Use your left hand for suggesting a root note and the right hand for suggesting a chord type.

# i Tip:

Live input can be recorded to a progression.

## **Related information**

Setting Up MIDI Input (on page 233) MIDI Routing (on page 41)

# **Recording Snippets**

The most common workflow is to grab **Phrases** from a **Library** and drop them on the **<u>Snippets Grid** (on page 24)</u>. You can however also record snippets directly.

- 1. Go to the **Structure** page.
- 2. Boot of the Snippet Grid View with the grid icon on the toolbar.
- 3. You can now select both, containers in the **Structure View**, or snippets in the **Snippet Grid**. Whatever you selected last gets input focus and shows up in the **Track Sheet**.
- 4. Double-click on a slot in the grid to create a new empty snippet.
- 5. Select the desired instrument in the track sheet.
- 6. Record a phrase (on page 260) as usual.
- 7. The snippet now includes the recorded phrase.

## i) Tip:

You can also record a phrase in any container of the arrangement and drop it on the snippets grid later.

#### **Related information**

Setting Up MIDI Input (on page 233) MIDI Routing (on page 41)

# Making a Sketch

## How to make and optimize a Sketch

## **Sketch From Container**

Select the desired container and do *Container > Make Sketch* to add a new Sketch to the *Playback* menu and select it for real-time playback immediately.

## **Sketch From Scratch**

Open a <u>Sketch App (on page 147)</u> with *File > New > Sketch* and add phrases and other parameters just as you would do with an arrangement.

# i) Tip:

If you want to save a temporary Sketch to a file, open it with *Playback > Edit Current Sketch* ... and do *File > Save As* ....

#### **Related information**

Sketch App (on page 147) Sketches (on page 32)

# Import, Export, Printing

About importing, exporting and printing documents

# Limitations Of Import

Synfire is able to read **Standard MIDI Files**. Import however, is actually a misnomer. Synfire is neither a DAW nor a MIDI file editor. Figure recognition (on page 5), the process of converting static MIDI data to dynamic phrases, is a lossy and ambiguous process. Moreover, MIDI tracks are full of repetition and duplication you need to sift through and eliminate manually, since this task can't possibly be automated without making the imported arrangement sound even more different from the original.

A more appropriate label would be "**Creating New Phrases From MIDI Files**", because that's exactly what it does. Standard MIDI Files do not include the necessary meta-data that would be needed to properly import an arrangement and convert it to phrases that exactly render their original MIDI notes. This is an impossibility, as is <u>explained here (on</u> <u>page 5)</u>.

That being said, harvesting phrases is still one of the best ways to build large libraries quickly. Apart from generating phrases randomly with KIM Factories, that is.

#### Important:

Import is a tool for harvesting useful **Phrases** from MIDI files and collecting them in a **Library**. Importing arrangements whole for the purpose of "editing" them is pointless.

### **Related information**

Import SMF (on page 264) Library App (on page 149)

## Import SMF

#### Video Tutorial

Before you begin, make sure you understand the <u>limitations of MIDI import (on page 264)</u>. We recommend you import into libraries, sift through the imported tracks and collect useful phrases.

## **Preparation**

If you export MIDI files from another program, or have a chance to edit them, make sure they are best prepared for import.

- 1. Files must be in Standard MIDI File Format (1 channel for each instrument)
- Program change messages should conform to the General MIDI Standard, so Synfire can automatically determine the kind of instrument for each track.
- 3. Drums and percussion should be on channel 10.
- 4. Only one sound per channel (no dynamic sound changes).
- 5. If possible, avoid upbeat, lead-in, or other events before the actual start of the piece. Likewise, nothing should come after the end.
- 6. Synfire can handle non-quantized music. However, quantization can be helpful to create cleaner and leaner phrases.
- 7. Notes must match the time signature and beat of a file. Synfire does not perform any after-the-fact tempo synchronization or beat recognition.
- 8. While key changes are recognized and handled, you should prefer files with a single key signature.
- 9. Remove any unnecessary and redundant data form a MIDI file if possible.

#### **Possible Destinations**

#### **Create a New Library**

Open a Library App with File > New > Library and do File > Import > Standard MIDI File.

#### Import Into The Embedded Library

In an **Arrangement App**, do *Library > Import > Standard MIDI File* to have all imported phrases added to the embedded library.

#### Import Into a New Arrangement

In an **Arrangement App**, do *File > Import > Standard MIDI File* to create a new arrangement based on the imported MIDI tracks.

#### Note:

If you want to re-import a MIDI file you have previously exported from Synfire, do *File > Import* > *Standard MIDI File (Copy My Sounds)* to copy the current arrangement rack over to the new arrangement. However, please consider the fundamental limitations of this approach (on page 264).

#### **Import a Single Phrase**

In an **Arrangement App**, select the desired target container and instrument. Then do *Phrase > Import* ... to import a single MIDI track into that phrase.

## **Import Settings**

After selecting a file for import, the **<u>Standard MIDI File Import** (*on page 204*) dialog pops up, allowing you to select the tracks you want to import and configure the handling and processing of MIDI data. The import process is then started on that dialog.</u>

## **Post-processing**

After an import has finished, you have the option to further improve the result.

- 1. Do the recognized chords somehow appear implausible? It is possible that Synfire incorrectly selected a parallel key. Try it yet again and confirm the detected key yourself.
- 2. Figures should spread out a bit around the center ("zero") line. If figures have been mistakenly set too far apart, transpose them manually by moving the entire vector seven steps (about an octave) upwards or downwards.
- 3. Do the segments seem to be torn apart and unusually messy? Try a new figure recognition with other settings: Select the Take parameter and perform the figure recognition in the inspector.
- 4. Play the arrangement. It should sound something like the original. If not, check once more whether all prerequisites for MIDI file import are met.

## **Re-Importing From a DAW**

If you exported the MIDI file from a DAW yourself and the file is laid out for a particular device with specific MIDI channels, you can proceed as follows in order to retain the file's track structure and sounds:

- 1. General MIDI Standard : Disable this, unless the device confirms to the GM standard.
- 2. Retain MIDI Channels : Tick this box.
- 3. Switch to the **Settings** tab and use the **Sound Wizard** to setup the sounds for all tracks. This creates a rack for the arrangement that is about to be imported.

#### Troubleshooting:

If harmony seems messed up, double-check if drum tracks or other non-tonal content was accidentally included with the harmonization.

### **Related information**

Import MIDI File Dialog (on page 204) Limitations Of Import (on page 264)

# **Export SMF**

Exporting in **Standard MIDI File** format (SMF) lets you send the finished composition or parts of it to a DAW or notation program. You have the option to export an arrangement as a whole, or only parts of it as outlined below.

### **Export Arrangement**

Use File > Export > Standard MIDI File ... to begin an export of the whole arrangement.

### **Export Container**

Use Container > Export > Standard MIDI File ... to begin an export of the selected container.

## **Export Track**

To export an entire track's worth of MIDI output, you can either <u>drag & drop the phrase (on page 270)</u> in the root container, or <u>drag & drop from a Drone's user interface (on page 268)</u> inside the DAW.

## **Export Phrase**

Single phrases can be exported with drag & drop (on page 270) only. Your last used export settings will apply.

### **Settings**

On the file dialog window that pops up, you can make a few settings to control what exactly is written to the exported file.

#### **Program Changes**

Add a MIDI program change messages to each track, if the sound has one.

#### **Controller Reset**

Add a message to reset all MIDI controllers to default values at the beginning of each track.

#### Lead-in

Insert a measure of pause at the beginning in order to leave sufficient room for MIDI controllers and program change messages to settle.

#### **General MIDI**

Use GM-compatible program change messages for each sound, corresponding to its Category.

#### **Optimize For Notation**

Quantize note positions and lengths and close small gaps between notes. This may improve import into a notation program.

#### **Global Key**

Estimate and write a single global key at the beginning, instead of writing multiple key changes to the file.

#### Chords

Export chord names as text elements.

#### **Container Structure**

Export container names as text elements.

# i Tip:

Synfire remembers these settings, so when you do a drag & drop export next time, they will be used.

#### **Related information**

Drag & Drop Export (on page 270) Export Notation (on page 268)

# **Export From Drone**

If you are working with **Drones**, you can drag a Drone's current MIDI content and drop it on a DAW track.

- From the arrange window, do File > Export > Export MIDI From Drones to DAW. This transmit the entire composition to all Drones. If you just saved the arrangement to disk, you need not do this.
- 2. Switch to the DAW and open the Drone you want to export from.
- 3. Drag the big rectangle in the left corner and drop it on the desired MIDI region in the DAW.
- 4. The drone switches to Play DAW mode after this. This means it no longer plays any music rendered by Synfire, but instead takes its input from the DAW track now. You can alter this mode on the <u>Rack Module (on page 131)</u> that controls the Drone.
- 5. Done.

### Note:

Be sure to drop exported MIDI data at the exact position that corresponds to where the song starts in Synfire. Otherwise playback will be out of sync.

# **Export Notation**

Synfire can write notation file formats that many notation programs are able to open. Originally, this export feature was intended for songwriters to print a lead sheet with chords, lyrics and melodies. It is however capable enough to assist you with notation tasks beyond that limited scope.

## **MusicXML**

MusicXML files can be imported by most notation programs. You should adjust the page layout and other options in your notation program before you print the score.

## LilyPond

LilyPond is a free open source program that produces printed notation of excellent quality. This is not a format for data exchange between programs, but rather a scripted language for printing scores. The files can be edited easily to change page layout and other options.

Please visit <u>https://lilypond.org</u> for more information. Download the latest version of LilyPond and install it on your computer. If you installed LilyPond correctly, Synfire will call it automatically and a PDF will be generated and opened.

In **Preferences** under the **Files** tab, you must set the location where LilyPond is installed. You may need to tweak the command line setting and look for errors in the LilyPond log to iron out issues if they occur.

#### **Preparation**

Before you export an arrangement for notation, make sure you understand the following.

- 1. **Global Key**: Consider that Synfire assumes the global key for the clef (key signature), which also determines all exported accidentals. If there are many key changes in your progressions that you don't want to see printed, tick the box **Global Key** in the global **Notation Export Settings**.
- 2. **Playing Ranges**: Make sure the playing ranges of your instruments match those of the real thing, if you export for orchestra or other natural instruments.
- 3. Voice Separation: Split multi-staff parts over multiple instruments (with the same sound) and check to have them all merged into the same staff on export with Notation Export Settings. This is not necessary, but has benefits.
- 4. **Note Positions And Durations**: Quantize Figure symbols exactly to what you want to see in the score. Be especially precise with triplets and dotted notes. Avoid swing or other forms of expressive rhythm. Make everything as simple as possible.
- 5. **Rests And Overlaps**: Use **Force Legato** everywhere unless the rests between notes are intended. Clean up unnecessary note overlaps.
- 6. Sustain Pedal: Remove this controller and disable Generate Pedal with Interpretation (on page 301).
- 7. **Morphing**: Freeze all morphed regions with a **Snapshot**, delete the <u>Morphing (on page 305)</u> parameter and clean <u>Figure (on page 294)</u> up by hand as detailed before.

## **Export Options**

Make sure you configure the global export options (on page 65) and options for each instrument (on page 61).

#### **Export Arrangement**

Use File > Export menu to export the whole arrangement.

#### **Export Container**

Use Container > Export to export the selected container only.

#### **Configuring External Editors**

In the global <u>**Preferences** (on page 190)</u> under **Notation**, you can configure three command lines for processing exported notation files.

# Note:

This feature is available with the **Pro** edition.

# Export Audio (Bounce To Disk)

Click the **Cassette Tape** button on the transport bar in order to arm Synfire for audio export. Press **Play** to start the playback. Audio output is now saved to disk. When playback has finished, you need to press **Stop** to close the exported file (otherwise it will continue capturing all audio output).

You will find your exported audio file next to the arrangement. Open the folder with File > Show in Finder (or: Explorer).

## Important:

All instruments must be hosted by a single Audio Engine for this to work.

# **Drag & Drop Export**

#### Video Tutorial

To export a **Standard MIDI File**, grab an <u>instrument name (on page 99)</u> or <u>container (on page 94)</u> and drag it over to a DAW or your desktop. Wait a few seconds for the mouse pointer to change to a **MIDI** icon before you drop it. Only after the delay will Synfire turn the internal drag and drop operation into an external one.

If you don't want to wait, you can also tap the space bar once you are hovering over the target in order to activate the external export.

### **Exporting Chords**

To export plain MIDI chords of any progression, grab the green **Harmony** outlet on the **Palette** page, or anywhere else where you find this outlet, and drag it to your DAW or desktop. The MIDI file will contain the chords as they are interpreted by the global **Chords** instrument.

You can use *File > Convert > Extract Progressions* to summon all progressions found in the arrangement and open a **Progression Editor** on them. This is especially worthwhile for songs whose chord changes you would like to share with other users.

## Note:

Containers always export with meta data (e.g. tempo, time signature and chord names). Individual instruments don't. If you need meta data without rendered music, you can export **Global Parameters** by dragging that label.

#### Note:

Synfire always uses the settings you configured for your last **Standard MIDI File** export. In order to change these settings, you need to export from the menu *File > Export > Standard MIDI File* before exporting with Drag & Drop.

# **Reusing Containers**

If you want to reuse a container of another arrangement (source) in your current project (target), this requires a bit of caution. Both arrangements may not have much in common, which prohibits a simple drag & drop. The following points need to be considered.

- 1. Transfer each phrase with copy & paste or drag & drop from the source to the target. If an instrument is yet missing, drop the phrase on the empty area or between the tracks of the **Track Sheet**.
- 2. Both arrangements may use different sounds. When you drop a phrase, Synfire will ask you whether to also copy the sound to the target arrangement, or use a different sound.
- 3. The source may be influenced by inherited parameters that are not present in the target. This is not a problem, but you should brace for possibly different results.
- 4. The source may have child containers. You need to recreate them in the target before you copy their phrases. Note that if those children are aliases, you must copy the phrases of their originals.
- 5. Check whether the source has Global Parameters that you are not aware of.

While you are doing this transfer, you may find yourself motivated by new ideas that lead to different results than you originally envisioned. Don't worry. If your goal was to exactly replicate the original, you could have as well edited a copy of it.

# Printing

Synfire can currently print **Palettes** and **Progressions**. The command *File > Print* ... is enabled only on the **Progression** and **Palette** pages of the **Arrangement App**.

The standalone Palette App and Progression App can also be used for printing.

Example 10	Syntire/PalettesJ Example 11	л (эку High).co	gpro* [7-Strin	ig ⊧lectric Gui	ar B-E-A-D-G	-в-е]
Ebm t - Eb.aeolian Eb.natural-t Eb.Minor	Gbmaj7 tP B tG, sP - Gb.ionian B.lydian Eb.natural-r Eb.Minor Eb.Minor	Ddim - D.altered-d r Eb.harmoni- Eb.Minor	Ebm t - Eb.aeolian Eb.natural-r Eb.Minor	Gbmaj7 tP - Gb.ionian Eb.natural-r Eb.Minor	B tG, sP - B.lydian Eb.natural-r Eb.Minor	Ddim - D.altered-de Eb.harmonic Eb.Minor
XXX	Gbmaj7					
Ebm Gbmaj7 B Ddim	m7 m(add9) m7sus4 maj7(add4) maj7(9) r (add2) 6 maj7 (add9 dim7	m9 m11 naj7(13) maj3 ) 6(add9) m	7(9,11) aj7(9) maj7(	13)		
Cognitone - S	Syntire Pro 2.0.8 - Page 1 o		otional Inform	nation	_	_
Print Scheme	<ul> <li>V Diagrams</li> <li>7-String Electric G</li> <li>Chord Sc</li> <li>All Octaves</li> <li>No Duplicates</li> </ul>	uit: 💙 💙	List Of Chords	s ions	Previ 1 V C	ew Page: Color Landscape Cancel

## **Palettes**

If you want to print a palette in black and white, use the coloring preset Coloring > Preset > Printing.

## **Progressions**

Progressions on paper can be helpful for a rehearsal with real instruments. You can prepare a printout for each musician and their string instrument or keyboard specifically.

#### Progression

Print the chord progression as it currently shows up in the progression editor. You can disable this if you only want to print the fingerings (tablature) or keyboard helpers.

#### **Helpers**

Print fingerings (tablature) or keyboard patterns. You may choose to print the scale instead of the chord.

Each Chord Once ensures each unique chord is only printed once.

#### **Optional Information**

Prints a list with additional information that can be useful when playing together and for improvisation.

# File Management

How to organize and handle your projects and files

## **Arrangement Templates**

When Synfire opens a new arrangement window, it looks into the folder Templates inside your **Configuration Folder** and presents the files in there as starting points for new work. By saving additional arrangements to that folder, you can extend this collection of templates to support your personal workflow.

## **Reverting To a Previous Version**

Synfire keeps backups of previous versions of your files in a sub-directory backups next to the file. The maximum number of versions to keep can be set with *Preferences*.

If you want to return to the last saved version and have not yet saved the currently open file, you can do so with *File* > *Revert to Saved*.

If you want to restore a previous version, look it up under *File > Backups* and select the version you want to restore. Doing so will merely open that file and not change anything on your hard drive until you save it.

## i Tip:

To free up space on your hard drive, you can delete all previous back-up copies of a file at once with *File* > *Remove Backups*. You may safely delete the folder backups altogether, but be aware that this will remove a lot of backups of several other files also.

#### **Archiving Projects**

Future versions of Synfire may potentially use new algorithms that produce slightly different output, as the rendering technology evolves over time. You should therefore not rely on Synfire documents as a long term storage format for your finished work. Be sure you export your work to notation programs, to a DAW or to Standard MIDI Files, before you upgrade to a new version of Synfire.

#### Note:

Although many random decisions are involved with the rendering process, Synfire always produces the same output for the same input parameters. This is because random streams get reset to a defined initial state before each run.

# Installation

## How to install or uninstall Synfire

#### **Upgrade From an Earlier Version**

Always <u>make a full backup (on page 276)</u> of your system before you install an update. You should do that at least daily anyway. At the very least, backup your current Synfire projects and other saved files.

#### Important:

It is not necessary to uninstall a previous version of Synfire before you install an newer updated version. During installation, all important settings will be migrated and preserved. An uninstall is only needed when you possibly need to revert to an earlier version.

#### iLok License Manager

Before you install or update Synfire, always download and install also the most recent version of **iLok License Manager** from the <u>PACE website ilok.com</u>. Synfire will not run without it, even if no specific license is required for a demo or other utility.

#### **Install Synfire On macOS**

- 1. Download the most current version of Synfire from your user account at the <u>Cognitone website</u>. Your web browser will place the downloaded archive in your <u>Downloads</u> folder.
- 2. Right-click on the package named Synfire Pro.mpkg (or similar) and select Open.

You will see a warning that the developer can't be verified, which however is merely because the package needed to be compiled on our servers prior to download and, thanks to Apple's lack of support for other operating systems, could not be signed properly on a Linux system. You can safely dismiss the warning, provided you obtained the download directly from our website.

- The macOS system installer will install the Audio Engine and all associated system components and documents and open another SETUP application that will guide you through the steps of installing Synfire itself.
- 4. When **SETUP** has finished, you will find the Synfire program in your **Applications** folder.
- 5. Please keep your original download files in a safe place. This will allow you to reinstall your current version of Synfire at any time.

#### **Install Synfire On Windows**

- 1. Log-in to Windows with an administrator account. Administrator rights are required to install Synfire. Otherwise Synfire will not be able to install or update required system components.
- 2. Download the most current version of Synfire from your user account at the <u>Cognitone website</u>. Your web browser will place the downloaded archive in your <u>Downloads</u> folder.
- 3. Extract the ZIP archive completely before you proceed. This is a required step. Launching SETUP.exe from inside the ZIP archive is not supported.
- 4. Please regard the file README.rtf, if available. It contains the most current installation instructions.
- 5. Launch the program SETUP. exe found among the downloaded files. If at some point you need to reinstall the **Audio Engine**, you can also do so by running Cognitone Audio Engine.msi. Installing Synfire however does not work this way.
- The Windows system installer will install the Audio Engine and all associated system components and documents and open another SETUP application that will guide you through the steps of installing Synfire itself.

- 7. When **SETUP** has finished, you will find a shortcut to the Synfire program on your desktop.
- 8. Please keep your original download files in a safe place. This will allow you to reinstall your current version of Synfire at any time.

### **Uninstall Synfire From macOS**

Apple does not provide a standard uninstall procedure. Instead, run the **Automator** script Uninstall Synfire Pro (or similar) that came with the original download. You will be asked for your log-in password, so the script can obtain administrator rights in order to be able to remove the system components associated with Synfire.

#### **Delete Files Manually**

If for some reason the above method does not work, you can manually delete the files associated with Synfire.

- 1. /Library/Application Support/Cognitone. If you have other Cognitone software installed, make sure you only delete the sub-folders associated with Synfire.
- 2. /Library/Application Support/Synfire
- 3. /Users/Shared/Documents/Synfire/Examples
- 4. /Applications/Synfire.app
- 5. All Drone plug-ins in /Library/Audio/Plug-Ins/Components, VST and VST3
- 6./Library/Application Support/Propellerhead Software/ReWire/ Transport.bundle

#### Files You May Want To Keep

If you potentially want to re-install Synfire at a later time, you should keep or backup the following files, which contain your personal configuration and documents. They are located under your user account's home directory.

- 1. ~/Library/Application Support/Cognitone.
- 2. ~/Documents/Synfire
- 3. Your custom **Projects** folder, if you changed it.

#### **Uninstall Synfire From Windows**

Synfire and the Audio Engine can be removed from the Windows **System Control Panel** in the **Apps** section. Alternatively, you will also find an Uninstaller in the Windows *Start* menu right under Cognitone. Administrator rights are required to uninstall Synfire.

#### **Delete Files Manually**

If for some reason the above method does not work, you can manually delete the files associated with Synfire.

- 1. C:\ProgramData\Cognitone. If you have other Cognitone software installed, make sure you only delete the sub-folders associated with Synfire.
- 2. C:\ProgramData\Synfire
- 3. C:\Users\Public\Documents\Synfire\Examples
- 4. C:\Program Files\Cognitone\Synfire
- 5. All Drone plug-ins in C:\Program Files\Steinberg\VstPlugins or any other directory you chose when you installed the Audio Engine
- 6. In the Windows registry, delete the key HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft \Windows\CurrentVersion\Uninstall\Synfire 2 using the command regedit from the command prompt as an administrator.

#### Files You May Want To Keep

If you potentially want to re-install Synfire at a later time, you should keep or backup the following files, which contain your personal configuration and documents. They are located under your user account's home directory.

- 1. C:\Users\<name>\AppData\Roaming\Cognitone.
- 2. C:\Users\<name>\Documents\Synfire
- 3. Your custom **Projects** folder, if you changed it.

# Backup

## Where to find the files and folders to include with a manual backup of Synfire

#### Important:

We strongly recommended you backup your installation of Synfire and the projects and files you created using your operating system's preferred tools and schedules. On macOS this is **Time Machine** and on Windows this is the backup utility found in **System Preferences**.

Should you ever need to manually backup or restore the files you have created with Synfire, this is where you find the relevant files.

## **Home Folders**

These are all found under your personal home directory.

#### ~/Documents/Synfire Documents\Synfire

The default folder for all your settings and projects. It should have a sub-folder for Arrangements, Sketches, Libraries, Palettes and other document types. We recommend you keep your saved files in this hierarchy in order to simplify backup and migration to future versions of Synfire.

Inside this folder you find the so-called **Configuration Folder** Config that contains your customized system preferences, **Device Descriptions**, **Palette Colorings and Layouts**, **Plug-in Presets** and more.

Your **Global Racks** also go here into the **Racks** folder by default, unless you save them to a more specific location.

## Note:

Version 1 used distinct names **Synfire Express** and **Synfire Pro** for this folder, which are now migrated into a single folder for all editions **Sparks**, **Express** and **Pro**. This is to ensure a smoother upgrade experience and simplify file management and backup.

~/Library/Application Support/Cognitone AppData\Roaming\Cognitone

This is where the **Audio Engine** stores its settings and the list of scanned plug-ins. You need not necessarily backup this folder, unless you manually patched individual files.

### **System Folders**

These are found under the root folder of your startup volume.

/Library/Application Support/Cognitone C:\ProgramData\Cognitone

This is where Synfire installs the apps of the **Audio Engine** and the SETUP application used to install or uninstall Synfire. You need not backup these manually, as they can be restored by re-installing Synfire.

/Library/Application Support/Synfire C:\ProgramData\Synfire

This is where user manuals, update logs and system-provided settings and defaults are stored in order to keep them separate from files you customized for your own purposes. It also includes a Configuration Folder the content of which you should not modify.

#### Note:

Version 1 used distinct names **Synfire Express** and **Synfire Pro** for this folder, which are now migrated into a single folder for all editions **Sparks**, **Express** and **Pro**. This is to ensure a smoother upgrade experience and simplify file management and backup.

#### **Making a Manual Backup**

Grab all of the above folders and copy them to a thumb drive, external disk or other data volume. In order to not confuse folders that bear the same name, be sure to create empty destination folders Home and System first before you copy the original folders into them.

#### **Restoring a Manual Backup**

Uninstall and re-install Synfire first. Then copy your backup folders back to their original locations as they are explained above, replacing any existing files.

# Chapter 5. Workflows

## Suggestions for approaches you may want to try

Composing music is a highly creative, chaotic and unpredictable process. Every songwriter, composer or producer develops their own personal approaches and habits over time. Synfire however is an entirely new tool to which most routines you have internalized with a DAW no longer apply.

Synfire supports new agile workflows that can be overwhelming for beginners. This section is about introducing you to general approaches of organizing a project and developing an arrangement. It will help you getting productive more quickly.

# Strategies

## General approaches to staying focused and productive

Creativity never follows a predictable path. Still, there are general strategies you can pursue to stay focussed and not get distracted by too many possibilities. The following sections outline different approaches.

## **Collect And Arrange**

The most convenient strategy is to assemble music from phrases and parameters you have already at hand in the form of libraries or previously composed arrangements. This requires a bit of preparation.

- 1. You begin with collecting phrases (on page 281) and progressions (on page 282) according to your personal
- preferences. Keep everything you like, for whatever reason, even if it is unlikely to be used anytime soon.
- 2. Once you have filled a few libraries, look for phrases that inspire you most for a new song.
- 3. Drop phrases on a new arrangement to add instruments, extend phrases, change harmony, etc.
- 4. Develop the arrangement following your intuition.

## i Tip:

You should do the collecting on a different day than the arranging. Collecting phrases is more routine and repetitive, while arranging is chaotic and intuitive. Both states of mind do not go together well.

## **Import And Rebuild**

You may have a number of DAW projects or MIDI files at hand that you want to harvest for musical expressions that worked well. This strategy is about reusing some of your previous work to create new music.

- 1. Export previous work as Standard MIDI Files.
- Import (on page 264) these files, preferably as static pitches to speedup the process. Make sure every track with a tonal instrument is included with the estimation of <u>Harmony (on page 300)</u>.
- 3. In the imported arrangement, skim every Take for interesting fragments. Select them with the **Span** tool and collect a phrase into the embedded library with **ℜE** Control-E or *Phrase > New Phrase From Selection*. This is explained in more detail here (on page 50).

- 4. Save the embedded library as a file with Library > Save As ....
- 5. Open the library on the sidebar of a new arrangement and use it to build the parts of a new composition.
- 6. Develop the arrangement following your intuition.

You can also import directly into a library using the standalone Library App (on page 149).

## CAUTION:

Be sure you understand the <u>limitations of MIDI file import (on page 264)</u> before you attempt to faithfully recreate previous work. This strategy is about rebuilding music from individual elements. You will almost certainly come up with a variation and reinterpretation of the original.

## **Concept And Execution**

The crucial part of this strategy is to start with a general idea that intrigues you and take that as a motivation to build something unique. Here are a few examples.

- The idea to have a song based on a single chord only with varying extensions.
- The idea to compose a modal ambient piece with slowly evolving phrases.
- The idea to use excessive ostinato (same note over and over).
- The idea to use chords from a chromatic palette and forget about traditional harmony.
- The idea to make excessive use of parameters together with extremely short figures.
- The idea to change time signature in the chorus.
- The idea to continuously move from beginning to end without any repetitions, in a single big chain of evolving counterpoint figures and rhythms, with a dramatic build-up towards a climax and a final resolution.

Whatever idea you come up with, the key to this strategy is a deep understanding of Synfire. It's definitely not for beginners, but will reward power users with unparalleled results.

## i Tip:

No matter how extreme or harsh an idea may sound at first, if you listen to the results a few days later, you will notice that it actually isn't that radical but quite refreshing. The key to great stuff is to be not afraid of extremes.

#### **Related information**

Editing (on page 240) Import SMF (on page 264) Collecting Phrases (on page 281) Collecting Harmony (on page 282) Building Structure (on page 284)
# Songwriting

### Arrange a song based on sections like verse, bridge and chorus

Songwriting is most productive when you follow the <u>Collect & Arrange (on page 278)</u> workflow. At the time you begin with a new song, you will already have a lot of phrases at hand that you can try out and see what works best with the lyrics and style you have in mind. The phrases will put any chord progression to live in an instant.

There are probably a million ways to come up with an idea for a new song. Playing around with Synfire is one of them. This workflow summary therefore focusses on how to arrange and refine a song based on existing phrases.

### **Song Parts**

You only need very few phrases for a song. Build a riff or pattern for the verse first and then make variations of it as you develop other parts, like bridge and chorus.

- 1. Create a **Container** for each part. Leave room at the beginning for one measure in order to simplify the synchronizing with a DAW later on.
- 2. Use an **Alias** where a part repeats later in the song.
- 3. Use Pause to turn individual instruments on or off while the song evolves. It is a good idea to keep all pause parameters in separate child containers, so you can use different combinations of pauses at different places in the song.

Another possible approach is to start with a very dense pattern for the chorus (the climax of your song with everything maxed out) and then strip it down and thin it out for the more mute parts like intro and verse. This ensures a consistent rhythm and makes the parts similar to each other, which is a great way to build up anticipation and tension. Jumping right into a completely different part may be too disruptive for unsuspecting listeners, but ultimately it depends on your style, how harsh or smooth you want it to be.

Yet another approach is to generate related but somewhat different parts with a **Factory** and glue them together. Make sure though that rhythm is compatible across parts. The best way to verify that is to get up from your seat and move.

### Melody

Keep in mind that 80% of a melody is rhythm. If you already have lyrics, that will largely determine the rhythm. Once you have a rhythm that works, do the following.

- 1. Record a Step parameter (on page 261) by tapping in the rhythm on your external MIDI keyboard.
- 2. Move that **Step** parameter to the **Grid** <u>outlet (on page 88)</u> of the phrase toolbar. Now you have a grid to draw along.
- 3. Draw a horizontal line with the **Line Tool** using the blue horizontal symbols. Be sure <u>duplicates are enabled (on</u> <u>page 88)</u> to get a series of multiple symbols.
- 4. Nudge individual Figure symbols up or down to shape a melody.

The more convenient method of coming up with a melody is to generate some with <u>Factories (on page 29)</u> and keep and refine those you like.

(this is not finished yet)

## **Collecting Phrases**

Phrases are the lifeblood of music, much more so than chords and sounds. Since it is impossible for Cognitone to meet everyone's demand with pre-manufactured libraries, given the vast variety of styles and individual preferences out there, collecting your own phrases is imperative if you want your music to stand out.

So, now that you became a hunter and gatherer in the realm of **Music Prototyping**, here are different ways to get hold of precious phrase material.

#### Drawing

Drawing a **Figure** in the **<u>Phrase Editor** (on page 247)</u> is the most flexible way to create a phrase, although it takes some patience and practice before you get productive with this.

#### Combining

A simple way to get new phrase material is to edit or combine already existing ones. Altering melodies comes to mind first. You can also isolate individual lines out of a phrase to use them standing alone. Or append multiple phrases to get one with more variation.

#### Recording

If you are good at playing the keyboard, there is no reason not to leverage that virtue for prototyping. Nothing could be more targeted and individual than a performance you recorded yourself.

You can record a **Phrase** directly into an arrangement when you need it, or use the <u>Library App (on page</u> <u>149)</u> to record and collect phrases in preparation for future projects. The actual recording task works the same in both cases, as <u>explained here (on page 260)</u>.

#### **Generating With Factories**

Synfire 2.0 introduced Factories (on page 29) that you can use to generate a huge variety of phrases for all sorts of purposes.

### **Agony Of Choice**

Factories are probably the most inspiring and surprising way to collect phrases, even though it is often a challenge to assess whether a particular phrase will eventually be useful or not. Still, you can't possibly keep them all. You have to delete a majority of them as soon as possible, or risk getting swamped in huge phrase pools that are impossible to sift through when you need them.

Polyphonic phrases with multiple parts are easier to assess than melodies or bass lines, because the latter often make sense only in context with other instruments. Over time you will learn to recognize a good melodic phrase, even if it sounds bland or generic in isolation.

#### **Harvesting MIDI Files**

Whether you recycle your own earlier projects, or search for files on the Internet, importing and disassembling **Standard MIDI Files** is a very targeted way to create phrases.

#### **Keep Phrases Short**

Remember that phrases are supposed to be flexible building blocks. You want to be able to combine them in new ways in order to achieve something new. Importing entire tracks is not only tediously slow, you also end up with long phrases full of repetitive content that is too clunky to be useful in a prototyping workflow. What you need is brief and unique fragments of musical expression that are no longer than 4 to 8 bars.

When you examine a MIDI file, look for the essence of each track. Keep only the spans with unique expressions and drop all repetitions and insignificant variations.

### **Related information**

Editing Parameters (on page 243) Editing Figure (on page 247) Editing Harmony (Progressions) (on page 250)

# **Collecting Harmony**

Pulling chords from a palette is intuitive and easy. Having a library full of progressions ready when you need them is even more comfortable. Here are a few tips how to collect chord progressions.

#### **Surfing Palettes**

Open a **Palette** and explore chord changes that sound they way you like. This is a very convenient way of finding new progressions. Just drag a chord off the palette and drop it into the palette's notepad view. When a progression is complete, you can grab the **Harmony** outlet and drop it into a **Library**.

#### **Transpose & Combine**

Unlike audio or static MIDI clips, the **Harmony** parameter can be easily transposed in order to work in every key. Just transpose a progression (or a part of it) up or down until the global key setting matches the key you need. Once they are transposed into key, partial progressions can be combined freely.

#### **Factories**

There are **Factories** that generate random **Harmony** parameters. Sometimes they are great to be used as a whole. Often they contain a few chord changes that inspire you. You can copy these portions into a progression editor and build a more meaningful harmonic narrative that suits your taste.

#### **Song Books**

The ability to enter one or more chords by name into the **Parameter Inspector's** input field is often overlooked. Grab a song book or a visit a website with songs and write down the chord names right into Synfire. With the Pro edition, you can also write chords in **Roman Numeral** (on page 323) notation or as **Harmonic Functions** (on page 319).

### **Related information**

Editing Harmony (Progressions) (on page 250)

# Production vs. Notation

Before you start a new arrangement, be clear about your goals, whether you want to export a score for notation (export MusicXML or LilyPond), or render a musical performance for audio production (export to DAW).

Notation export requires strict quantization (note onsets and durations) and a rather minimalistic arrangement.

For audio production, you probably want to maximize rhythmic expression and variation and enhance the overall listening experience by all sorts of edits and enhancements that will look overly complex and confusing in exported notation.

### **Related information**

Export Notation (on page 268)

# Audio Engine vs. DAW

Although Synfire offers great flexibility through the use of **Drones** and the synchronization with your DAW, we strongly recommend you begin your first projects based on the **Audio Engine** alone. Your arrangements will be wholly selfcontained and restore all audio setup automatically when opened. This is especially convenient when you open multiple arrangements at the same time in order to exchange phrases, sounds and other data with drag & drop.

- 1. Synfire is in control and behaves much like its own DAW.
- 2. No back and forth between two programs. Plug-ins can be opened and edited in Synfire.
- 3. No conflicts accessing shared MIDI ports or audio drivers (Windows).
- 4. No keeping of separate files for DAW and Synfire. No need to take care of when to open and save which files.
- 5. Multiple arrangements can be open at the same time.

Once your composition or song reaches a certain maturity, you can relocate it to a DAW for production. **Drones** then host all plug-ins on behalf of (and controlled by) Synfire. You can still replace sounds and edit your entire composition, while the audio processing is already done by the DAW.

Only after you exported all rendered MIDI physically to DAW tracks and unloaded the Drones, your project will become final and all further editing constrained by the capabilities of the DAW.

### **Latency Playing Drones Live**

The synchronization with a DAW for regular playback is almost perfect (sample precision). Latencies may occur when playing a Drone live through an external MIDI keyboard, though.

It takes some time for the MIDI messages sent by Synfire to arrive at the Drone and an additional time elapses until the DAW is processing them. The latency compensation built into most DAW unfortunately does not go into effect when playing a Drone this way. This is especially an issue with recording.

### i Tip:

In order to minimize the latency, you can connect your external keyboard to the DAW directly and arm the track with the Drone for monitoring. Please make sure you disable input for MIDI keyboard at the Synfire end or you will receive all input twice.

## **Building Structure**

While most projects start out with all phrases in the root container, sooner rather than later you will need to build a structure of nested containers.

### **Container Roles**

Containers take on different roles and serve different purposes in an arrangement.

- **Parts**: The most common role for a container is to represent a section or part of a song. That's where you usually put phrases with a Figure parameter.
- Landing Containers: After some climax, crescendo, build-up or chorus, a landing container introduces relaxing contrast by pausing several instruments for a few beats.
- Aliases: A previous section reoccurs, possibly modified by a parallel container that alters a few things to make it sound different this time around.
- Mixing: Use the Pause parameter in an empty child container to enable or disable individual instruments for the duration of that container.
- (this is not finished yet)

### **Tips for Containers**

Containers need not necessarily contain figures. There are many ways you can use other parameters, sometimes to dramatic effect. Most importantly, containers can be convenient tool for non-destructive editing and experimentation.

- Abstract Parameters: Populate containers not only with Figure, but with other parameters that influence their rendering: Transpose, Step, Variation, Velocity, Pause, Shift and many others.
- **Temporary Harmonies**: Throw a **Harmony** into an empty container and move the container to different places. You will be amazed at the things that will happen!
- Variation Of Aliases: Make an Alias of a container and move it to another location. Place a new container next to it with *Container > New Parallel Container* and fill it with parameters you want to override in the alias.
- Synchronization of Rhythm: Put a Step into the Global Parameters of a container and hear all instruments play that rhythm in sync. This is very effective for a short dramatic interplay.
- Continuous Morphing: Put a Morphing into the Global Parameters of the root container. All transitions between all containers will then be influenced by the morphing settings for all instruments.

(this is not finished yet)

# Chapter 6. Tutorials

Instructions and links to videos and examples online

# Palette Surfing 1

For beginners, it is helpful to do a few simple exercises to get used to the layout of Palettes.

### **Simple Cadence**

Visit the **Palette** page and select a **Standard Palette** in Minor from the **Circle of Fifths**. Now recall a preset *Layout* > *Preset* > *Harmonic Functions* for a layout that shows harmonic functions instead of chord names.

Select a **Sketch** from the *Playback* menu that puts some emphasis on the chords, for example, *Pop Push w/Guitar*.



Don't get confused by the symbols. All you have to know now is that <u>r</u> stands for *Tonic*, <u>b</u> for *Dominant*, and <u>s</u> for *Sub-dominant*. The lowercase letters are for minor, and the uppercase are for major. These chords are also called the **Primary Functions** of a key, according to the <u>Theory of Harmonic Functions</u> (*on page 319*).

Now play the sequence

t - S - D - t

It will probably sound very familiar to you because this is a **Cadence**. Cadences clearly indicate to a listener which key we are in. This is most obvious with an **Authentic Cadence** that passes from  $\underline{D}$  to  $\underline{T}$  or  $\underline{E}$ , or in Roman Numeral notation from  $\underline{v}$  to  $\underline{I}$ , also written  $\underline{v} - \underline{I}$  or  $\underline{5} - \underline{1}$ .

### i Tip:

As a rule of thumb, a V - I authentic cadence is always a good way to close a progression, or establish a new key after a key change.

## Palette Surfing 2

### **Substitutions**

You can add variation to a progression by substituting chords by functionally related chords that sound different. One hint to find these replacements is their harmonic function.

According to the <u>Theory of Harmonic Functions (on page 319</u>), secondary functions can be derived from primary functions by appending the letters P, P (Parallel) or G, G (Gegenklang). Lowercase letters stand for Minor and uppercase for Major.

When you play the sequence below in a palette, you will notice there are multiple chords for the same function that you can choose from.

tP - sP - dP - t

The fundamental cadence *Tonic* – *Sub-dominant* – *Dominant* – *Tonic* is still maintained, even when the original chords are replaced with chords that have a similar harmonic function.

# Palette Surfing 3

### **Secondary Harmonies**

Now that starting with the tonic t,  $\tau$  and returning to it via a  $v - \tau$  is a safe haven of sorts, you can take further risks and experiment with **Secondary Functions**. There are no fixed rules.

Here is an example:

t - d - dP - S - t - dP - D - T

## Palette Surfing 4

### **Key Changes (Modulation)**

Visit the **Palette** tab and make sure the sidebar with the **Circle Of Fifths** is shown. The major keys are on the outside, and their parallel minor keys are on the inside.

Now select the key of c major and switch to a layout Layout > Preset > Default to see the names of relevant chords.

The circle of fifths window always shows you in which keys the chord currently being played has a particular function. These keys are highlighted in darker colors and are also labeled with the function.

Now play the chord Gm. You obtain the picture at the right. According to this picture, the chord is

- 1. the dominant of C minor,
- 2. the tonic of G minor,
- 3. the sub-dominant of D minor and D major, and
- 4. the dominant of C major.

This information is extremely useful when you want to change the key. Let's assume that so far you have played the following chords:

C Gm ... ?

and now you would like to change to the key of D Minor. Now our chord in the palette appears at another position, namely, the sub-dominant.

You may remember the cadence that will confirm the new key: Play the authentic cadence v - I, preferably with a seventh chord (A7 and Dm). You can clearly hear that you have arrived in D Minor.

The progression is thus as follows:

C Gm A7 Dm

For fun, you can now do the same thing again and pretend that the last chord  $p_m$  is not the tonic, but rather in turn the sub-dominant of another key in order to jump into the next key with an authentic cadence according to the formula s - p - T. The circle of fifths tells you that  $p_m$  is the sub-dominant of A major and A minor. If you decide on major, click on A Major and continue with the cadence v - T.

Meanwhile, the complete progression looks like this:

C Gm A7 Dm E A

Once again? The same formula. This time we go to E Minor via B7 - Em. The result:

C Gm A7 Dm E A B7 Em

You have now changed the key three times with only eight chords in total. If you like, you can continue with the same formula until you finally come to c Major again.

### i Tip:

There is an example progression **Circular Modulation** composed according to this formula. You can open it with the Progression Editor.

# Chapter 7. Troubleshooting

### Solutions to problems and confusion that may arise

Occasionally you may wonder why your arrangement behaves in a certain way and not in another, or why you don't hear anything or don't hear what you expect. Synfire provides several troubleshooting options and information about the structure of your arrangement.

### **Report Potential Problems**

Use *Help > Potential Problems with Arrangement* to obtain a report on conspicuous findings collected during the rendering process. If your arrangement behaves in a puzzling way, this report may provide hints where to look for an explanation or solution.

### **Parameter Hints**

Use *View > Parameter Hints* to display explanatory information in the <u>Parameter View (on page 144)</u> about where a parameter is inherited, paused, empty or otherwise special in a not immediately obvious way.

### **Console Log**

Open the console with Window > Console for possible clues about unexpected events or errors that have occurred.

### **Online Help**

On our website <u>www.cognitone.com</u> in the **Support** area, you will find various FAQs and a link to the online version of this manual, which may be more current.

## **Graphics Glitches on Windows**

### Condition

The moving playhead leaves trails on screen, gray lines appear inside content when scrolling a view. Fonts do not display correctly. This happens on Microsoft Windows with high DPI monitors only.

### Cause

Odd DPI scaling factors like 175% or 225% cause misalignments through rounding issues. Synfire has problems addressing quarter pixels, so to speak. We have not yet found a solution that works with all DPI settings.

### Remedy

- 1. The glitches go away when you set your monitor's DPI scaling factor in the Windows **System Settings** to 100, 150, 200, 250 or 300 percent.
- 2. Alternatively, you can disable HiDPI for Synfire altogether by editing its desktop shortcut: Append the command line option **-H** after the program name that is written in quotes.

# Plug-ins Can't Access Sounds

### Condition

Some or all plug-ins loaded by the Audio Engine can't seem to be able to load sounds, presets and other contents.

### Cause

With macOS Big Sur and later, programs are required to ask the user for permission to access external volumes and other folders. Some audio plug-ins are not aware of this when they attempt to open their sounds and presets. Unless the Audio Engine has been granted permission, the plug-ins it has loaded don't have that permission either.

### Remedy

- 1. Open Audio/MIDI Setup and visit the Audio tab.
- 2. Select the Audio Engine that is unable to load sounds, if necessary.
- 3. Click on Grant File Permission
- 4. Point the file dialog to a volume or folder where the sound s are stored.
- 5. Reload the rack with the problematic plug-ins, or restart Synfire.

# State Of Drones Not Restored After Opening

### Condition

After opening in your DAW a project with Drones or MIDI Drones in it and then opening the associated arrangement in Synfire, the rack modules on the **Sounds** page are not restored to the state that was present when the arrangement was last saved.

Synfire shows a warning message that some sounds could not be restored when the arrangement is opened.

### Cause

Your DAW might not have loaded the Drone plug-ins, or not early enough.

### Remedy

Check the project settings in your DAW and make sure all plug-ins are always loaded, whether the DAW thinks they are currently needed or not.

# Unpredictable Auto-Chords Output

### Condition

Rendered MIDI output of the <u>Auto-Chords (on page 301)</u> feature changes significantly after making only minor progression edits or changes to the instrument's playing ranges.

#### Cause

<u>Auto-Chords (on page 301)</u> is a convenience feature that automatically populates a track with sustained chords of the current progression. It doesn't support detailed control of inversion, voicing and rhythm.

### Remedy

Disable <u>Auto-Chords (on page 301)</u> and draw individual chords using the <u>Figure (on page 294)</u> parameter to match the rhythm you have in mind. Draw chords by sketching vertical lines with the Line tool.

## Sound Allocation Monitor

In order to track down routing problems or unexpected MIDI behavior, you can open a sound allocation monitoring panel with *File > Sound Allocation Monitor*. This information can be helpful to fix MIDI-related issues and to understand the general sound management of Synfire.

Global Instruments										
G	Global Rack									
[E	[Engine1:01] Soundcase FluidR3									
1	мс	h i	Usage	Grp	Sound		Client / Instrument			
	- 1			0	Yamaha Grand Piano		Global Instruments > "Piano"			
	1	.1		0	Yamaha Grand Piano		Arrangement > "Yamaha Grand Piano"			
	1	.2		0	Yamaha Grand Piano		Global Instruments > "Yamaha Grand Piano" (Setup)			
	2			0	Synth Strings 1		Global Instruments > "Chords"			
	2	.1			Synth Strings 1		Global Instruments > "Chords" (Preview, Chords)			
	3				Acoustic Bass		Global Instruments > "Bass"			
	3				Acoustic Bass		Global Instruments > "Bass" (Preview, Bass)			
ê 🗖	4				Steel String Guitar		Global Instruments > "Guitar"			
	5				Standard		Global Instruments > "Drums"			
	5				Standard		Global Instruments > "Metronome"			
-	6				Bright Yamaha Grand		Arrangement > "Eight Bars Around The Block"			
[8	ingin	e1:02	] MA	STER: I	iZotope Ozone 6					
Containers Global Instruments [4/1] 1 "Piano" 2 "Chords" 3 "Bass" 4 "Guitar" 5 "Drums" 6 "Metronome" Preview [16/1] 1 "Chords" (Preview, Chords) 2 "Bass" (Preview, Bass) 3 "Yamaha Grand Piano" (Setup)										
Dancing Elephant Containers Dancing Elephant [16/1] 1 "Rhodes" <null> 1 2 "Yamaha Grand Piano" <null> 3 "Pad" <null> 4 "Bass" <null> Preview [4/1]</null></null></null></null>						<null> <null> <null> <null></null></null></null></null>				

### Note:

This feature is available with the Express and Pro editions.

# Chapter 8. References

Appendix with tables, lists and other resources

# **Encyclopedia of Parameters**

### Complete list of all parameters, their meaning and purpose

### Analysis

This parameter is used internally by the Harmonizer to store its results and preferences. You can delete it if you need to reset those to defaults.

### Chromatic

Transposes final MIDI output absolutely in chromatic steps. To preserve harmonic integrity, transposing in increments of 12 halftones is safe. Other increments may still lead to interesting results.

### **Properties**

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.

### Note:

Do not transpose output by octaves when you can alter the **Playing Range** of the **Instrument** instead. If for some reason the octave range of your device is extremely off, try compensating for this at the plug-in or synthesizer first.

### Note:

This feature is available with the **Pro** edition.



### Controllers

Several parameters resemble their standard MIDI equivalent. Values range from 0 to 127 or -64 to +64.

Some controllers like <u>Volume (on page 318)</u> and <u>Pan (on page 292)</u> are mapped to knobs and faders on the <u>Matrix</u> <u>Console (on page 104)</u>.

### **Properties**

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.

### **Custom Controllers**

User-defined <u>Custom Controllers (on page 199)</u> can be configured per <u>Sound (on page 37)</u> or <u>Device (on page 36)</u>. Their customized names will appear in <u>Parameter Outlets (on page 45)</u> where the sound or instrument is currently selected.

### Important:

The faders for <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u> and similar controllers don't do *audio* mixing. They merely transmit MIDI messages and set the corresponding <u>Parameter (on page 2)</u> that is used to render MIDI output. This has no effect if a plug-in or device doesn't respond to the MIDI controller as expected. The master faders and **Solo**, **Mute** and **Master** buttons work at the audio level, though.

### **Related information**

Custom Controllers (on page 199)

### **Dynamics**



**Dynamics** modulates the final <u>Velocity (on page 300)</u> of the output. Values are floating point numbers which the velocity gets multiplied by. Thus 1.0 doesn't change anything and greater values increase velocity while smaller ones damp it down. Since the scaling is proportional, natural variation is preserved.

Use this parameter to conduct and fine-tune overall dynamics without altering velocities.

### **Properties**

Using the fader (slider) on the <u>Parameter Inspector (on page 54)</u> sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the <u>Parameter</u> <u>View (on page 144)</u>.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to estimate a dynamics profile for the current Figure and drop it somewhere else.

Instead of drag and drop, you can also use copy and paste.

The parameter can be extracted from the current <u>Figure (on page 294)</u> with right-click menu or **Parameter > Extract** From Figure.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.

### Note:

This feature is available with the Pro edition.





### Video Tutorial

The Figure (on page 294) parameter carries a parametric format of musical expression (on page 2) that is independent of pitch and harmony. It contains **Symbols** grouped into **Segments**, the latter of which are responsible for preserving melodic movement and expression, regardless which harmony a Figure is rendered against.

### Important:

The most important thing about Figure is that it's not a substitute for notation or piano rolls. It's an **algorithm**.

### Note:

The term **Figure** stands for a technical data structure here. Not to be confused with the musical notion of a melodic shape.

### Segments

One or more **Symbols** are grouped to form a **Segment**. Segments denote musically meaningful units that are supposed to preserve their relative shape. Symbols in a segment may be stacked vertically (chord), placed in a row (melody), or any combination of the two.



Segments are selected, edited and transformed as a cohesive unit, making them a convenient device for building melodies and musical expressions of arbitrary complexity.

### Anchor

Every segment has one **Anchor** that denotes a point of emphasis that marks the melodically most important symbol of a segment. The anchor is rendered first and then other symbols are rendered relative to it. It also determines the position of the segment on the timeline.

### i Tip:

If you want a segment to end on a particularly important target note, make that last symbol the anchor. Likewise, if you care most about any other note in between, make it the anchor.

### **Symbol Types**



Symbols are notated on a staff with horizontal lines that, for most symbol types, correspond to the steps of a scale. The bold middle line is a reference that depends on the type of symbol.

Н

**Horizontal**: Best for long running melodies relative to the current key. The zero line denotes the root of the **Reference Scale** (on page 11) of the **Global Key**.

V

**Vertical**: Best for ornaments, licks or improvised melodies. The root of this scale changes with every chord. The zero line denotes the root of the <u>Vertical Scale (on page 8)</u>.

С

**Chord / Arpeggio**: Best for chords, horn stabs, arpeggios and finger picking patterns. The zero line denotes the lowest note of the current chord in its current inversion and voicing. Each line represents a note in the chord.

I

**Interval**: The anchor describes an interval of the chord. All other symbols are mapped relative to the anchor. Use this type to start a segment on a desired chord interval and let it continue from there.

For example, 3 maps to the third, 5 to the fifth, and 7 to the seventh (since the grid starts at zero, it is one off). Negative values go in the opposite direction. Since -3 is a third below root, it is no longer actually a third. We recommend you write positive anchors to avoid confusion.

#### Ρ

**Pitch**: Absolute pitch. For drums, percussion and other sounds that need not follow harmony. The zero line denotes C4.

#### В

**Bass**: The zero line denotes the bass of the current <u>Harmonic Context (on page 12)</u>. Each line is a step on the Vertical Scale.

#### R

**Relative**: The Anchor is mapped like a **Chord** symbol, while additional symbols are rendered like **Vertical** symbols relative to the anchor. Use this type to start a segment on a desired chord note and let it continue from there.

#### Gray

Exclusively used by the Take (on page 312) parameter. Absolute pitch.

The best way to make yourself familiar with the different symbol types is to draw a segment in a short **Figure** and let it repeat over and over while a chord progression moves on.

### Note:

Segments may contain only one type of symbol. Types cannot be mixed within the same segment.

#### Note:

The **Vertical** type is available with the Express and Pro editions. The **Relative** and **Interval** types are available with the Pro edition only.

#### Important:

Figure has no idea of pitch and thus can't be sight-read or played. It would be pointless to print it. It is also not meant as a substitute for MIDI or traditional notation. It is exclusively a tool for creative composing.

#### Important:

Although there are +/- seven lines in the grid, there are many scales that have more or fewer steps. The interval 0-7 is thus not necessarily always an octave. If you want to notate an octave, please use a chromatic shift of 12 or -12 for a symbol.

### **Positioning Of Segments**

The position of a segment in a **Figure** is determined by its anchor. It is therefore only included with a Span selection if the selection also includes the anchor. Likewise, a segment will not be rendered if its anchor is outside the bounds of a container.

Individual symbols of a segment may also occur at negative positions, that is, before the start of the parameter. Provided the anchor is inside container bounds, such a segment starts playing even before the container begins. This is useful for melodic lines that are supposed to pick up early before moving into a container, or finish gracefully after exiting a container.



Similarly, segments that reach beyond the end of a parameter will continue playing into the next loop or the next container.

### Tip:

i

Segments reaching across container bounds are only useful for very short overlaps. If you want a melody to overlap more than a bar, consider placing them in a container that is shifted on the timeline accordingly.

### Note:

The range of negative positions is limited. Symbols can't occur more than 4/4 before zero. The segment will also show a warning, if the anchor is at a negative position, which will exclude it from being rendered at all.

#### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Instead of drag and drop, you can also use copy and paste.

Right-click and Extract From MIDI Output to insert the current output as a figure with absolute pitch symbols.

### Accidentals

Symbols have **diatonic** and a **chromatic** component. The diatonic component determines the line on which the symbol is drawn. The chromatic component determines the number of semitones to shift away from that. Although this chromatic shift is called an **accidental**, its meaning is different from tradition notation.

Figure symbols always snap to the current scale or chord. Whether that note is traditionally notated with an accidental (flat or sharp) or not depends on the key.

Notes may land out of scale only if <u>Voice Leading (on page 73)</u> is bypassed and chromatic shifts are enabled for the segment (Pro). Such chromatic shifts can be pretty harsh. After all, the job of a scale is to make sure all instruments behave in a harmonically consistent manner. The introduction of blue notes or passing tones is a matter of picking an appropriate scale, if only for the duration of a single chord.

The chromatic component of a symbol means: If there is a note in the scale this number of halftones away, use it. So if you have a green chord symbol with a chromatic component of -1, there may be a note in the scale a halftone below that chord tone. If not, the next note in the scale below that chord tone is used.

### **Related information**

Rendering (on page 18) Figure Inspector (on page 71)

### Flow



Flow modulates the length of notes to achieve a variety of rhythmical articulations. Values range from -12 ... 12. For best musical suitability, the value range is split into distinct zones with different interpretations.

Val	Effect	Amount	Of
12	Overlap	1/16	Fixed
11	Overlap	1/32	Fixed
10	Overlap	1/64	Fixed
9	Overlap	1/128	Fixed
8	Legato	100 %	Step between onsets
7	Legato	90 %	Step between onsets
6	Legato	80 %	Step between onsets

5	Legato	70 %	Step between onsets
4	Legato	60 %	Step between onsets
3	Legato	50 %	Step between onsets
2	Legato	40 %	Step between onsets
1	Legato	30 %	Step between onsets
0	Original	100 %	Original length
-1	Shorten	90 %	Original length
-2	Shorten	80 %	Original length
-3	Shorten	70 %	Original length
-4	Shorten	60 %	Original length
-5	Shorten	50 %	Original length
-6	Shorten	40 %	Original length
-7	Shorten	30 %	Original length
-8	Shorten	20 %	Original length
-9	Staccato	1/16	Fixed
-10	Staccato	1/32	Fixed
-11	Staccato	1/64	Fixed
-12	Staccato	1/128	Fixed

The splitting of zones is a compromise to balance math and music. It affords us a parameter with symmetric range that can be easily sketched, inverted and converted from other parameters, without risking harsh musical outcomes.

### **Properties**

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main *Parameter* menu.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to estimate a flow pattern for the current Figure and drop it somewhere else.

Instead of drag and drop, you can also use copy and paste.

The parameter can be extracted from the current <u>Figure (on page 294)</u> with right-click menu or **Parameter > Extract** From Figure. The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.



### Harmony

В	F#	Gmaj7	A	В	Em7	Gmaj7	F#7	
1		2		3		4		1
								>
вт	<b>F#</b> D	<b>Gmaj7</b> tG, sP (t)	A dP, SS	вт	Em7 s(tGsP)	<b>Gmaj7</b> tG, sP (t)	<b>F#7</b> D	1
- B.ionian B.major B.Major B.Major	- F#.mixolydian B.major B.Major B.Major	- G.lydian B.natural-minor B.Major-minor B.Major	- Alydian B.major-b7 B.Major B.Major	- B.ionian B.major B.Major B.Major	- E.dorian+4 B.harmonic-minor B.Major-minor B.Major	- G.lydian B.natural-minor B.Major-minor B.Major	- F#.bebop-minor B.major-add#7-add7 B.Major B.Major	

The **Harmony** parameter holds a <u>**Progression** (on page 14)</u> that provides a <u>**Harmonic Context** (on page 12)</u> for every point on the timeline. In cooperation with the <u>Interpretation (on page 301)</u> parameter, it guides the rendering of <u>**Figure** (on page 294)</u> and other parameters to produce final MIDI output.

You can modify Harmony at any time, causing the entire Arrangement (on page 19) to be re-composed in an instant.

### **Properties**

Harmony supports **Polytonality** (on page 15), the superimposition of multiple chords at the same time.

The parameter is **Global** (on page 23) and thus influences all instruments in the container.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main **Parameter** menu.

### Conversion

Drop a **Container** or **Snippet** on this outlet to extract the parameter.

Drop a Harmony or Preview parameter to insert its progression here.

Instead of drag and drop, you can also use copy and paste.

Converting **Harmony** from **Take** is done by **Figure Recognition**, which is an interactive procedure that requires your attention.

#### **Related information**

Fundamentals (on page 6) Harmonic Context (on page 12) Harmony Settings (on page 69) Progressions (on page 14) Controlling Inversion, Voicing and Octave (on page 254)

### Interpretation

The <u>Interpretation (on page 301)</u> parameter guides the rendering of <u>Figure (on page 294)</u> and other parameters to MIDI. In particular it handles conflicts with respect to voice leading and <u>Harmony (on page 300)</u>.

### **Properties**

The parameter is **Constant**. It takes effect for the entire duration of a <u>Container (on page 21)</u>. If you need to change it in the middle of a container, insert a child container and modify the parameter there.

The parameter adds **Output** (on page 305) to the **Parameter View** to give visual feedback on its effect.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main **Parameter** menu.

### **Related information**

Interpretation Settings (on page 78) Fine Tuning Interpretation (on page 258)

### Auto Chords

### Generating sustained chords and bass automatically

**Auto-Chords** and **Auto-Bass** are convenience features of the <u>Interpretation (on page 301)</u> parameter that render sustained chords and bass notes automatically without needing a <u>Figure (on page 294)</u>.

Chord inversions, voicing and alignment set in the Harmony (on page 300) parameter are considered.

If there is already a Figure (on page 294), it will be rendered in addition to Auto-Chords and Auto-Bass.

Enable **Tie Notes** to sustain all notes that are already playing in the previous chord. The initial setting for a new interpretation is assumed from *Playback > Tie Auto-Chord Notes*.

### i) Tip:

To convert Auto-Chords into a Figure (on page 294), drag Harmony (on page 300) to the Figure (on page 294) outlet. Edit the figure's segments to customize inversions, voicing and dynamics they way you want.

### i Tip:

To control the overall pitch range, adjust the middle **Playing Range** of the instrument and its **Interpretation** (on page 301) settings.

### Layer



This parameter supports **Polytonality** (on page 15) in that it assigns one of up to four variations of the Harmony (on page 300) parameter to individual instruments. Each such layer may use a different chord, effectively superimposing them over other chords at the same time.

### **Properties**

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.



This feature is available with the Pro edition.

### Length



With the Length parameter, you can override (replace) the note lengths of a figure.

#### **Properties**

The parameter is **Polyphonic**. It can contain multiple values at the same position.

The parameter can be edited in Hyper Edit mode.

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

#### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to grab the lengths of the current Figure symbols and drop them somewhere else.

Instead of drag and drop, you can also use copy and paste.

The parameter can be extracted from the current <u>Figure (on page 294)</u> with right-click menu or **Parameter > Extract** From Figure.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.

### CAUTION:

Length is now deprecated in favor of Flow (on page 298), because absolute lengths do not translate well to different phrases.

### Note:

This feature is available with the Express and Pro editions.

### Lyrics



You can use the Lyrics parameter to associate syllables, words, and other text with symbols of a figure. In exported notation files, these are rendered underneath the notes.

For reference, the **Figure** (on page 294) is displayed in the background. Active the **Snap To Figure** option to make text input snap to positions with symbols automatically.

Text is entered in the **Text Input** field on a sentence-by-sentence basis, with individual syllables separated by hyphens (-). Spaces are not allowed within a single word. The underscore character (\_) serves as an extender. It stands for an empty (silent) syllable. You need extenders to preserve rhythm where the previous syllable continues to sound over multiple notes in succession.

Examples:

High a-top the yel-low wa\_a-gon

### She is so\_\_\_ in-cre-di-bly beas-ty and in-sa-tia-ble

Lines of text, as well as individual syllables, can be moved (transposed) from one line to another using the arrow keys.

This parameter is **Polyphonic**. It may have up to four values at the same point in time, representing lines of text placed above each other. Printed for **Text Input**, the syllables arranged above each other are enclosed in parentheses.

### i Tip:

You can select a span of multiple syllables and edited them selectively. If the total number of syllables does not change, their previous positions are retained.

### i Tip:

It is more convenient to set each line of text independently and then bring them together via the clipboard with *Edit > Paste and Merge*.

### Note:

This feature is available with the **Pro** edition.

### Morphing

The <u>Morphing (on page 305)</u> parameter allows for a **Phrase** to seamlessly cross-fade into the phrase in the next <u>Container (on page 21)</u>, creating a continuously transforming phrase along the way.

#### **Properties**

The parameter is **Constant**. It takes effect for the entire duration of a **Container** (on page 21). If you need to change it in the middle of a container, insert a child container and modify the parameter there.

The parameter adds Output (on page 305) to the Parameter View to give visual feedback on its effect.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main *Parameter* menu.

## Note:

This feature is available with the **Pro** edition.

### **Related information**

Morphing Setttings (on page 82)

### Output



This parameter shows generated MIDI output.

It is useful for visual feedback about the effect of parameter changes. The <u>Overview (on page 100)</u> page uses it to offer you a bird's eye perspective on your score.

Notes are color coded.

### Magenta

Note matches the current bass, which may or may not be included with the chord.

#### Green

Note belongs to the chord.

### Blue

Note belongs to the vertical scale.

#### Red

Note is a chromatic alteration not in the vertical scale.

### Important:

This parameter is **read-only**. Although you can select and inspect all notes, you can't edit them. This wouldn't make sense because any parameter change anywhere in the arrangement will make your edits obsolete as soon as all output is generated anew from scratch.

### Pause



The Pause (on page 306) parameter is convenient for temporarily suppressing the output of a Phrase.

Pause is musically sensible in that it does not break <u>Figure (on page 294)</u> segments apart. Whether a segment is audible is determined by the position of its <u>Anchor (on page 295)</u>.

A value of 1 means output is paused, while a value of 0 means the phrase is playing.

### **Arranging With Pause**

The track number of a **Track Header** offers a shortcut to the <u>Pause (on page 306)</u> parameter. Click on the track number multiple times to switch between paused, playing and neutral.

This way you can enable or disable an instrument for the duration of a <u>Container (on page 21)</u>. By placing various containers with <u>Pause</u> (on page 306) parameters in them, you effectively control which instruments are playing when and for how long. This is a great non-destructive way to build up tension and is a lot more flexible than deleting or inactivating phrases.

### i Tip:

You can use <u>Pause (on page 306)</u> to thin out dense phrases, or control the interaction of multiple phrases with each other by alternatively pausing the other instrument.

#### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.

### Preview

**Preview** is equivalent to <u>Harmony (on page 300)</u>. It is a placeholder for a preview progression to use when rendering phrases where no <u>Harmony (on page 300)</u> parameter is otherwise available.

You can define a default preview progression in Preferences > Defaults.

#### **Properties**

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main *Parameter* menu.

#### Conversion

Drop a Container or Snippet on this outlet to extract the parameter.

Drop a Harmony or Preview parameter to insert its progression here.

Instead of drag and drop, you can also use copy and paste.

#### **Related information**

Harmony (on page 300)

### Rhythm



The <u>Rhythm (on page 307)</u> parameter modulates note positions and velocities according to a rhythmic profile. It is relatively similar to what is known as 'groove quantization' with some DAWs and sequencers.

Unlike the <u>Step (on page 310)</u> parameter, note positions and velocities are altered only gradually. This parameter exerts a certain feel of rhythm rather than doing a strict quantization.

#### Name

A label that might be useful if save a rhythm profile as a template.

#### Thumbnail

Rhythm is usually a short loop of 1m or 2m length. The thumbnail illustrates the effect it has on note positions and dynamics.

### i Tip:

Assign a <u>Rhythm (on page 307)</u> to the **Global Parameters** of a <u>Container (on page 21)</u> to have all instruments play the same groove.

### **Properties**

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main **Parameter** menu.

### Conversion

Drop a Figure (on page 294) on this Outlet to analyze its rhythmic structure and insert a rhythm profile.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to estimate a rhythm profile for the current Figure and drop it somewhere else.

Instead of drag and drop, you can also use copy and paste.

Right-click and Extract From MIDI Output to estimate a rhythm profile from current output.

### Scheme

The term **Metric Scheme** (short: Scheme) refers to a combination of time signature and verse metrics. It breaks down the continuous stream of time into rows of a certain number of measures. This adds 'line breaks' to a **Progression** (on page 14) that correspond to the verse structure of a song.

Scheme is conveniently set by picking a time signature from the preset menu. Verse metrics can be altered on the **Scheme** tab on the **Progression** page. When you assign a new time signature, previously existing verse metrics are reset to defaults.

### **Progressions**

Internally, a <u>Scheme (on page 308)</u> is associated with every <u>Progression (on page 14)</u> in order to remember its intended time signature and verse metrics. The scheme of a progression however is superseded by the current <u>Scheme (on page 308)</u> of the container you put it in.

### **Settings**

These settings provide hints for future interpretation, but are not currently used.

### Half Time

Only every other beat is emphasized, resulting for a slower feel.

#### Straight vs. Swing

Whether rhythm is suggested to have a swing factor.

### **Properties**

The parameter is **Global** (on page 23) and thus influences all instruments in the container.

The parameter is **Constant**. It takes effect for the entire duration of a **Container** (on page 21). If you need to change it in the middle of a container, insert a child container and modify the parameter there.

### Conversion

Drop a **Container** or **Snippet** on this outlet to extract the parameter.

Instead of drag and drop, you can also use copy and paste.

### **Related information**

Progressions (on page 14)

### Shift



The <u>Shift (on page 309)</u> parameter alters note positions to play earlier or later. It lets you achieve subtle to dramatic effects, depending on its amplitude. From a gentle "humanizer" to expressive rubato effects (steal-time), everything is possible. In the extreme, entirely distorted phrases can be created.

### **Properties**

The parameter is **Polyphonic**. It can contain multiple values at the same position.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main **Parameter** menu.

Using the fader (slider) on the <u>Parameter Inspector (on page 54)</u> sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the <u>Parameter</u> <u>View (on page 144)</u>.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Instead of drag and drop, you can also use copy and paste.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

### Note:

This feature is available with the Express and Pro editions.

### Skip

The **Skip** parameter determines how many positions of the <u>Step (on page 310)</u> parameter are to be skipped at any point on the time line.

The parameter is intended to be used with multiple instruments that follow the same <u>Step (on page 310)</u> parameter but are supposed to play a different subset of the given steps.

Values are positive integers including zero.

This parameter requires a Step parameter to be present in the same container to have any effect.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Instead of drag and drop, you can also use copy and paste.

### CAUTION:

The parameter is not officially supported and is considered experimental.



This feature is available with the **Pro** edition.

### Step



The <u>Step (on page 310)</u> parameter re-arranges the positions of all symbols in the Figure parameter sequentially from left to right. Results can be dramatic and surprising, as the rhythm of a phrase is altered.

This parameter is always looped. However, if the Figure is not looped, it will be rolled out only once.

### **Properties**

The parameter can be edited in Hyper Edit mode.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main *Parameter* menu.

#### Conversion

Drop any non-constant parameter to insert the positions of its values as steps.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to grab the steps of the current Figure and drop them somewhere else.

Instead of drag and drop, you can also use copy and paste.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.

The parameter can be extracted from the current <u>Figure (on page 294)</u> with right-click menu or **Parameter > Extract** From Figure.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

### *i*) Tip:

Put a <u>Step (on page 310)</u> parameter in **Global Parameters** to make all instruments play the exactly same rhythm for the duration of the container.

### i) Tip:

Have multiple instruments play the same Figure (on page 294), but set a different Step (on page 310) for each instrument.

Note:

This feature is available with the Express and Pro editions.

### Sustain



The Sustain (on page 312) parameter resembles the standard MIDI controller for the sustain pedal.

Values can be 1 (pedal pressed, notes keep sounding after key is released) and 0 (pedal released, notes stop when key is released).

The Interpretation (on page 301) parameter has a setting to generate Sustain (on page 312) pedal controllers automatically, depending on Harmony (on page 300) and phrase properties.

### Conversion

Drag this outlet to generate an auto-pedal for the current Figure and drop it somewhere else.

Instead of drag and drop, you can also use copy and paste.

The parameter can be extracted from the current <u>Figure (on page 294)</u> with right-click menu or **Parameter > Extract** From Figure.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

### **Related information**

Interpretation (on page 301)

### Take



The <u>Take (on page 312)</u> is a temporary buffer for <u>recording (on page 260)</u> and import. It contains the last recorded take or imported MIDI file track in the form of absolute pitches (MIDI). It provides the input for subsequent **Figure Recognition** or the **Harmonizer**.

#### Note:

You can create variations of an already rendered phrase by using *Parameter > Extract from Output* and then recycle that Take (on page 312) by recognizing it again.

#### Conversion

Drop a Figure or Output to insert their current pitches as a take. Better yet, do a right-click and *Extract From Output*. This way the current segment properties are preserved in the form of tags.

#### **Related information**

Take Settings (on page 74) Recording (on page 260) Recording Parameters (on page 261)

### Tempo

**Tempo** is a floating point parameter with values denoting beats-per-minute (BPM). Synfire supports continuous tempo changes.

### **Tempo Tap-in Feature**

Using the **Parameter Inspector** (on page 54), you can tap ¼ notes on your external MIDI keyboard. Press the **Tap-In** button and start tapping until the tempo value stabilizes. Finish by clicking the button again.

### i Tip:

With *File > Convert > Scale Tempo* you can globally double or halve the tempo of an arrangement. All phrases are then automatically stretched or compressed to reflect the change.

### **Properties**

The parameter is **Global** (on page 23) and thus influences all instruments in the container.

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop a Container or Snippet on this outlet to extract the parameter.

Instead of drag and drop, you can also use copy and paste.

### Time



This parameter is used for film scoring.

Unlike all other parameters, the horizontal axis of <u>Time (on page 314)</u> is not measured in bars and beats, but absolute time. Cue markers therefore dynamically change position depending on <u>Tempo (on page 313)</u>. Which is the purpose. You place **Cue Markers** at absolute times in order to see where they occur in the music.

This allows you to <u>fine-tune (on page 259)</u> <u>**Tempo** (on page 313)</u> such that a particular beat in the song meets with a moment in time.

### **Properties**

The parameter is **Global** (on page 23) and thus influences all instruments in the container.



### Note:

The time being displayed is measured from the beginning of the container in which the parameter is set. In order to avoid confusion when scoring for film, you should therefore set the <u>Time (on page 314)</u> parameter in the root container.

### Note:

This feature is available with the **Pro** edition.

### **Related information**

Time Inspector (on page 84) External Synchronization (on page 66) Fitting Music Between Cue Markers (on page 259)

### Transition

This parameter is only relevant for sketches. It determines at which locations Synfire should be prepared for a possible chord change during real-time playback. Only at these locations will Synfire switch to the next chord chosen by the player.

Two templates can be called:

- 1. From Scheme: Use the current metric scheme
- 2. **From Current Progression**: Adopt the positions of the cords from the current progression. You can use this to configure even unconventional rhythms.

You are free to set the positions manually with the phrase editor.

### **Properties**

The parameter is **Global** (on page 23) and thus influences all instruments in the container.

### Transpose



Transposes the <u>Figure (on page 294)</u> parameter before it is finally rendered. Its unit is scale steps and values can be integers or floating point numbers, possibly creating accidentals as a result.

Each symbol is transposed individually, while the melodic shape of segments is preserved where possible.
### i Tip:

With Synfire, transposing music is much more relaxed than with conventional sequencers. You need not worry about potential dissonances at all and may use it as a means for introducing variation and expression.

### **Properties**

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.

### Variation



The <u>Variation (on page 316)</u> parameter controls which of a set of predefined transformations are applied to <u>Figure</u> (on page 294) segments before they are rendered.

Values are floating-point numbers between 0.0 and 1.0 mapped to a total of 25 transformations (e.g. Invert, Flip, Transpose, Stretch, Reverse, etc).

### **New Variation**

Click to try a new random value for the parameter.

### Thumbnail

Illustrates the level of randomness, although individual transformations can't be distinguished here.

There is little in the way of achieving a specific outcome, so this parameter is more of a heuristic "surprise me" feature intended to kick off new ideas. If you like a result however, you can do *Parameter > Save To Figure* to permanently apply the transformation to the **Figure** and clear the **Variation** parameter.

### **Properties**

The parameter adds Output (on page 305) to the Parameter View to give visual feedback on its effect.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Instead of drag and drop, you can also use copy and paste.

The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

## Velocity



The <u>Velocity (on page 317)</u> parameter overrides the velocities of <u>Figure (on page 294)</u> symbols. It can be used to transfer the dynamics of one phrase to another, or to alter dynamics over a time span that is longer than the figure itself.

### **Properties**

The parameter can be edited in Hyper Edit mode.

The parameter is Polyphonic. It can contain multiple values at the same position.

The parameter supports custom templates. You can save and recall **Parameter Templates** from its right-click menu or the main **Parameter** menu.

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range.

Drop a Figure, Take or Output to extract this parameter and insert it here.

Drag this outlet to grab the velocities of the current Figure and drop them somewhere else.

Instead of drag and drop, you can also use copy and paste.

The parameter can be extracted from the current Figure (on page 294) with right-click menu or **Parameter > Extract** From Figure. The parameter can be applied to the current Figure with the right-click menu or *Parameter > Save To Figure*, which will permanently alter the figure and clear the parameter when done.

Right-click and Extract From MIDI Output to insert what can be derived from the current output.

### i Tip:

Try a <u>Velocity (on page 317)</u> that is longer than the <u>Figure (on page 294)</u> to achieve a dynamic build-up, variation or development over time.

### i Tip:

Transfer the dynamics of one phrase to another by copying the <u>Velocity (on page 317)</u> parameter with drag & drop.

# i Tip:

Record the parameter <u>Velocity (on page 317)</u> with your MIDI keyboard in order to change the dynamics of a phrase more intuitively.

### **Related information**

Recording Parameters (on page 261)

### Volume



Resembles the standard MIDI controller. Values range from 0 to 127.

The controllers is also mapped to a fader on the Matrix Console.

### Important:

The faders for <u>Volume (on page 318)</u>, <u>Pan (on page 292)</u> and similar controllers don't do *audio* mixing. They merely transmit MIDI messages and set the corresponding <u>Parameter (on page 2)</u> that is used to render MIDI output. This has no effect if a plug-in or device doesn't respond to the MIDI controller as expected. The master faders and **Solo**, **Mute** and **Master** buttons work at the audio level, though.

### **Properties**

Using the fader (slider) on the **Parameter Inspector** (on page 54) sets a particular value for the entire duration of the container (constant). If you want to continuously automate this parameter, you need to draw values in the **Parameter View** (on page 144).

### Conversion

Drop any parameter with a curve-like shape. The form is preserved as far as possible and values are scaled into range. Instead of drag and drop, you can also use copy and paste.

# Harmonic Functions

### Brief introduction to the theory of harmonic functions according to Hugo Riemann

The theory of harmonic functions can be traced back to the German music theorist Hugo Riemann (1849–1919). It was revised by Herrmann Grabner (1886–1969). This theory is still taught today at German music schools. The theory is not without controversy and by no means a natural given. Synfire makes use of this theory only to the extent that it may be useful and inspiring for creating chord progressions.

### **Basics**

The theory describes a system of relationships between harmonies that spread out around a Tonal Center (root). Letters (function symbols) are used to identify the harmonies:

```
T, t = Tonic
S, s = Sub Dominant
D, d = Dominant
P, p = Parallel
G, g = Gegenklang ("contrast" in German)
N = Neapolitan
```

On the one hand, when resolved in relation to a tonal center, these symbols point to a particular major or minor triad. On the other hand, they also represent the idea of a "function", in the sense of a role or purpose they serve inside a chord progression.

Apart from the primary functions **Tonic**, **Dominant**, and **Sub Dominant** and their immediate relatives, Synfire does not speculate whether there may be some hidden magic behind higher level functions. We are looking at this more from a practical perspective, as an aid for understanding and creating progressions, not unlike the well-known **Roman Numeral** notation.

### **Primary Harmonies**

The primary harmonies of a key are **Tonic**, **Dominant**, and **Sub Dominant**. The Dominant is a fifth above the Tonic and the Sub Dominant a fifth below. Because each is a fifth away from the root, we say there is a **Fifth Relationship** (German: "Quint-Verwandtschaft") between primary harmonies.

### Tonic

The **Tonic** is the triad with the root on the first degree of the **Horizontal Scale** of the key. Thus, the root of the chord is also the root of that scale. A major triad is denotes as a capital **T**, and a minor triad is written with a lowercase **T**. For example, if C is the root of our key, the following applies:

T = Ct = Cm

The Tonic is perceived as a center of calm, invoking a feeling of stability and rest. If a melody at its end comes down to a final conclusion that feels like "Amen" or "Om", then this last note is usually the **Tonal Center**, the root note of the Tonic. In classical music, the final chord in a minor key's progression is often replaced by its major version for a well-known effect (German: "Trugschluss").

### Dominant

The **Dominant** is written as **D** or **d**. It is on the fifth degree of the **Horizontal Scale**. In contrast to the Tonic, the Dominant is full of suspense and tension, yearning for a resolution towards the Tonic. It is often played with dissonant extensions to make this yearning more obvious.

D7 t D9 t

This resolution  $\mathbb{D}_{\rightarrow}$  T is called **authentic cadence**, also known as 5 1. It works best with a major Dominant chord, although Dominants may be minor as well. If you don't know how to conclude a chord progression  $\mathbb{D}_{\rightarrow}$  T is always a good option.

### Sub Dominant

The **Sub Dominant** is written as s or s. It is built on the fourth degree of the **Horizontal Scale**. In the tonal center of C (major or minor), the following applies:

```
D = Gd = GmS = Fs = Fm
```

In a chord progression, **Sub Dominant** chords (and the secondary harmonies derived from them) often precede a more dissonant **Dominant** chord, which then leads back to the **Tonic**.

### Neapolitan

This merry friend is notated as **n** and called the **Neapolitan Sixth Chord**. It is a major triad on the lowered second degree (minor second) of the **Horizontal Scale**. It can be used in place of the **Sub Dominant** and is often resolved towards the **Dominant**.

### **Examples**

Here is an example of the popular progression  $t \le D T$  resolved in three different keys:

```
Cm Fm G C
Am Dm E A
Ebm Abm Bb Eb
```

### **Secondary Harmonies**

**Secondary Harmonies** are built in relation to the primary harmonies. They are in a **Third Relationship** (German: "Terz-Verwandtschaft") to the primary harmonies. Their root is one third away from the root of a primary harmony. The harmony based on a major third away is called **Gegenklang** ("contrast" in German). The one based on a minor third away is called **Parallel**. They are notated g and g or p and p respectively, appended to the primary function:

tP, tp, tG, tg, TP, Tp, ..., dP, dp, dG, ..., SG, Sg

For example, the symbol tp stands for Tonic Parallel and pg stands for Dominant Gegenklang.

In order to keep it simple, we don't want to dig into the secondary harmonies any deeper. As a practical rule of thumb, replacing primary harmonies with their secondary relatives and vice versa is a thing you should definitely try.

### **Interlude Harmonies**

Functions may temporarily refer to a shifted Tonal Center. These are called interlude harmonies. It's a matter of opinion whether this already constitutes a key change (modulation) or just adds more tension to the mix (i.e. more accidentals). It much depends on the duration of the shift and whether the other Tonal Center is reaffirmed strongly enough. For example, a D T cadence (in the shifted key) might establish it as the new Tonal Center.

Synfire allows for text input of interlude harmonies using the official Riemann notation. All chords that refer to the shifted key are grouped in parentheses, while the root note of the chord immediately after the closing parenthesis determines the shifted Tonal Center:

t d (s DG) D T t d (s DG) [D] T

Thus the two chords (s - DG) above refer to the Tonal Center at the root note of D (after the closing parenthesis). That chord may be set in square brackets (as in the second example), to prevent it from sounding. Nesting multiple interlude harmonies can lead to interesting progressions that wander through several keys.

### **Distant Relationships**

**Dominants** may be chained (stacked) to build **Secondary Dominants**. The root of one Dominant is taken as the tonal center of the subsequent Dominant to obtain the **Dominant of the Dominant**. The same goes for Sub Dominants.

The notation is simple: DD is a double dominant, DDD a triple dominant, etc. The same goes for ss, sss, etc. These are also called **Dominant Chains**. Two examples in the Tonal Center of A:

```
DDDD DDD DD D = Db F# B E
S SS SSS SSSS = D G C F
```

Dominant chains achieve a great effect when they are understood only in hindsight, i.e. leading to a conclusion that is obscure enough until finally resolved with the last chord. Theoretically, secondary harmonies also allow themselves to be chained (stacked) to denote more distant relationships:

tGG, TPp, ..., spp

Although Synfire can deal with chained expressions to an unlimited extent, it is doubtful whether extremely remote relationships achieve anything that is perceived as more sophisticated or interesting (acoustically) than a much simpler expression.

### **Mixing Major And Minor**

As you may have noticed, a **Tonal Center** has no gender. It's neither major, nor minor. When working with harmonic functions, it is common for both major and minor variants of the same chord to occur in the same key, even though the chord's notes fall outside the **Horizontal Scale** of the key. You should not worry about that at all, because the unlimited mixing of major and minor offers more freedom and adds color and tension to your music.

### **Ambiguity**

Chords take on different roles (functions) depending on the context in which they are played. It probably comes as no surprise that multiple function expressions may resolve to the same chord, especially when expressions are nested.

If function symbols are shown in a **Palette** (on page 26), you will see many chords with multiple function expressions next to them. A few expressions show up in parentheses (not to be confused with interlude harmonies!). Example:

F6(add9) in A = sP tG (s)

Chord F6 (add9) has two functions SP, tG in A and a third function shows in parentheses, meaning it is a function of only a sub-triad of the chord (if the chord can be decomposed into multiple superimposed triads). Whether a sub-triad (subset) of a chord really constitutes a functional relationship is debatable. However, it is certainly interesting in a practical sense when you are looking for a substitute for a chord with a similar harmonic function but a different timbre.

### **Syntax**

With the Pro edition of Synfire you can input chords and progressions in harmonic function expression format. The original notation introduced by Riemann didn't include specific about the interval structures to use for each chord. Synfire recognizes all chord interval structures in the **Catalog** to be appended as a suffix to the expression following a colon.

The major or minor triad implied by the original Riemann expression is replaced with whatever chord expression you append after a colon. Merely the root note is preserved in that case. For simple chord extensions that can be written as a number, the colon is optional.

An optional bass interval can be appended after a slash.

t tP/5 TG:(9,13)/7

```
DP7
DDD:dim7
sg:m7(b5)
Tp:7sus4(no5)
[sG]
```

# **Roman Numerals**

### Popular notation for chords and progressions

Roman numerals can be used to notate chords and chord progressions independent of a key. You have the option to display roman numerals instead of chord names in a **Palette**. The Pro edition of Synfire also supports writing chords and progressions as text input in this format.

### **Syntax**

Component	Examples
Accidental	b
	#
Scale Step, Gender	Major: I II III IV V VI VII VIII IX X XI XII
	Minor: i ii iii iv v vi vii viii ix x xi xii
Other Interval Structures	Diminished: dim, ø, o
	Augmented: aug, +
	Other: maj7, (add9), 13, m7b5
Bass	/b5, /6, /7

### **Examples**

V

The triad implied by the original expression is replaced with whatever chord name you append after a colon. For simple chord extensions that can be written as a number, the colon is optional.

```
#iii
viø, vio, vi:dim (all equivalent)
iiv°, iiv+, iiv:aug (all equivalent)
bVII/#11
III:maj7
#II6
#ii:m6
```

i∶m7b5/b5

bVII:(7,9,13)/5

### **Related information**

https://en.wikipedia.org/wiki/Roman\_numeral\_analysis

# **Parameter Conversion**

### Table showing which parameters convert best to other parameters

Parameters are designed to be <u>copied and moved around (on page 242)</u> to allow for unlimited experimentation. When a parameter is dropped on a destination outlet with a different type, its data is converted to that type. For example, if you drop a <u>Figure (on page 294)</u> on a <u>Velocity (on page 317)</u> outlet, the velocity values of the symbols in that figure are extracted.

Drag	Drop On	Conversion
Figure	Velocity, Length, Step	Extracts velocities, lengths or steps of symbols
	Rhythm	Estimates a rhythm profile of the figure
	Flow	Estimates a flow profile from the figure
	Dynamics	Estimates a dynamics profile from the figure
	Sustain	Estimates sustain pedal activations from the figure
	Take	Converts the current output of the phrase to a Take
	Controllers	Converts the melodic shape of the figure to CC values
Harmony	Figure	Creates chord segments for each chord change
Take	Figure	Creates a figure with static pitch symbols
Parameter that resem- bles a shape or curve	Parameter that resem- bles a shape or curve	Converts the shape or curve as best as possible
Container	Harmony, Tempo, Scheme	Grabs the parameter from the container and inserts it

Only certain conversions make practical sense . These are listed in the table below.

For the <u>Figure (on page 294)</u> parameter and many others, you can also use the <u>right-click</u> menu on a parameter outlet to do things like **Save To Figure**, or **Extract From Figure**, or **Extract from Output**.



### CAUTION:

Conversions not listed here are considered experimental and may change in the future.

### CAUTION:

Note that the opposite direction often makes no sense. For example, dropping <u>Step (on page 310)</u> on a <u>Figure (on page 294)</u> won't apply the steps to the figure.

# Syntax

How to read and write chords, scales and lengths (durations).

### **Note Syntax**

Notes begin with an uppercase letter A, B, C, D, E, F, G. These can be followed by one or more accidentals indicating a raise (sharp) or lowering (flat) by one halftone:

```
D# = D sharp
Eb = E flat
Gb = G flat
F# = F sharp
F## = F double sharp
```

The C chromatic scale is written this way:

C, C#, D, D#, E, F, F#, G, G#, A, A#, B

Synfire assumes your MIDI equipment is tuned with well-tempered **12TET** tuning (Twelve-tone Equal Temperament), which is the default for all current hardware and software. Thus D# and Eb are assumed to send the same MIDI note. Still, Synfire presents and accepts the different enharmonic spellings correctly, depending on context.

Hence, the Db Major scale is written this way:

```
Db, Eb, F, Gb, Ab, Bb, C
```

### Note:

Synfire optionally supports German note spelling where H is rendered for B and B for A# (*Edit > Preferences*). Actually there is no reason to do so, unless you are doing a formal music education in Germany. This weird notation stems from a historical confusion of b with h. It is not used anywhere outside German speaking countries.

### **Interval Syntax**

The chromatic distance between two notes is called an **Interval**. Examples are minor third, augmented fifth, diminished seventh, etc. For the labeling of chords and scales, Synfire uses a shorter form, though, as listed in the column **Interval** below.

```
Semi- Note
tones
```

Interval

Names

0	С	1	Prime
1	C#/Db	b2	Augmented Prime / Minor Second / Half Tone
2	D	2	Major Second / Whole Tone
3	D#/Eb	b3	Minor Third / Trisemitone
4	E	3	Major Third
5	F	4	Perfect Fourth
6	F#/Gb	b5	Augmented Fourth / Diminished Fifth / Tritone
7	G	5	Perfect Fifths
8	G#/Ab	#5	Augmented Fifths / Diminished Sixth
9	А	6	Major Sixth / Diminished Seventh
10	A#/Bb	7	Augmented Sixth / Minor Seventh
11	В	#7	Major Seventh
12	С	8	Octave
13	C#/Db	b9	Minor Ninth
14	D	9	Major Ninth
15	D#/Eb	#9	Augmented Ninth
16	E	b11	Diminished Eleventh
17	F	11	Perfect Eleventh
18	F#/Gb	#11	Augmented Eleventh
19	G		Perfect Twelfth / Tritave
20	G#/Ab	b13	Minor Thirteenth
21	А	13	Major Thirteenth
22	A#/Bb	b14	Minor Fourteenth
23	В	14	Major Fourteenth

# Note:

This interval nomenclature assumes 12TET tuning (Twelve-tone Equal Temperament)

### **Chord Syntax**

Note:

The Pro edition of Synfire supports additional syntax for chord input that recognizes **<u>Roman Numerals</u>** (on page 323) and **<u>Harmonic Function Expressions</u>** (on page 319).

For the designation of chords, Synfire uses the standard North American notation commonly used for Jazz. The chord name always begins with the name of the root note, whose spelling depends on the key in which the chord is used.

The root is followed by the designation of various triad forms (or nothing at all, if it is a major triad). For example, an m for a minor triad, dim for a diminished triad, aug for an augmented triad, and so forth:

Am, Cdim, F#aug, G, Esus4

An optional numeral 6, 7, 9, 11, or 13 stands for sixth, seventh, ninth, eleventh, and thirteenth chords:

Am9, C7, Gm9, F#13, Bmaj7, Ebmaj9

Extensions may be appended. These additional notes are numerals, optionally prefixed with  $\frac{1}{2}$  or  $\frac{1}{2}$ , listed in parentheses and separated by comma:

A7(9,#11), Cm7(b9), Emaj7(9,11), Am(7,9,13)

If only a single extension is added, an alternative notation uses the keyword add:

A(add9), Cmaj7(add4)

Many chords allow for multiple equivalent notations, although only certain notations are commonly used. For example, these chords on each line are identical:

```
Am9 = Am(7,9)
Am11 = Am(7,9,11) = Am7(9,11)
C13 = C(7,9,13)
Faug = F(#5)
Fmaj7 = F(#7)
Fmaj9 = Fmaj7(9) = F(#7,9)
```

In practice, the exact notation you choose for text input is not relevant in Synfire, as chords will be renamed automatically.

The standard chords included with the <u>Catalog (on page 16)</u> are shown below. You can add more chords to the Catalog as you need.

Dim.	Minor	Major	Augm., Susp.
Cdim	Cm	С	Caug
Cdim7	Cm6	C(add2)	Csus2
C(b5)	Cm7	C6	Csus4
C6(b5)	Cm7(b5)	C7	C7sus4

### Synfire User Manual | 8 - References | 328

C7(b5)	Cm9	C7(b9)
	Cm(maj7)	C7(b9,#9)
	Cm6(add9)	C9
	Cm7(#5)	C7(#9)
		Cmaj7
		Cmaj7(9)
		C(add9)
		C6(add9)

### Note:

Chord symbols must not include white space. All letters, digits, parentheses and numerals must be written without gaps. Multiple chord symbols however can be separated by spaces.

### **Slash Chords**

**Slash Chords** are written with a bass note appended after a slash. The bass note need not necessarily be a member of the chord.

Am/F# C/A

### **Power Chords**

**Power Chords** omit the third interval, playing only the root note and the fifth. The power chord is an interpretation of the major or minor triad. It cannot be added to a progression directly, because it has no name in the **Catalog**. Writing F(no3) doesn't help either, because Synfire needs to consider the full triad to ensure harmonic consistency for all instruments.

If you want a particular instrument to play power chords, use the **Chord** symbols of the **Figure** parameter to draw a chord with only two symbols for the prime and fifths.

### Warning:

Resist the temptation to add a power chord interval structure to the Catalog, e.g. cs. Using this in a progression would force **all** instruments to use only the two notes of the power chord, which is certainly not what you want.

### Scale Syntax

Like chords, scales begin with the name of their root note. A period follows that separates the root note from the scale's name, which is arbitrary (i.e. not parsed like chord symbols):

Eb.hungarian-minor C.major F#.aeolian

The name may be followed by a hyphen and references to features, such as added, altered or omitted notes. Accidentals and alterations use #, - or +, - respectively:

```
F.altered-dominant-bb7
E.locrian+2
C#.lydian-augmented
B.natural-minor-b2
```

The character @ followed by a digit says the object in question is the n<sup>th</sup> inversion (or rotation) of the scale. The example below denotes G natural minor, starting from the fourth degree, or Mode 4 of natural minor:

G.natural-minor@4

A dot followed by  $\underline{h}$  at the end (not unlike a file extension) denotes a **Horizontal Scale** that was automatically generated by Synfire from a **Vertical Scale**:

```
F.bebob-minor@3.h
blues1.h
```

In the course of your work with Synfire, you will actually never be confronted with having to input scales. The program makes these decisions for you automatically.

### **Scale Set Syntax**

<u>Scale Sets (on page 6)</u> always start with a capital letter. Otherwise, they are written like <u>Scales (on page 6)</u>. If you create your own, you are free to assign them any name.

### **Syntax of Durations And Times**

Several inspectors in Synfire allow for text input of time offsets and lengths (durations). These are notated as fractions, denoting a note length in a format that is easily understood by every musician. The shortest supported duration is 1/128. Durations shorter than that are denoted as MIDI ticks (see below).

1/1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128

With dotted lengths, each dot (asterisk) adds ½ of the previous length. Up to three dots are supported.

```
1/4*, 1/8**, 3/4***
```

### Triplets

1/3, 1/6, 1/12, 1/24, 1/48, 1/96

### Quintuplets

1/5, 1/10, 1/20, 1/40, 1/80

Of course triplets and quintuplets may be dotted, too. Odd tuplets like 1/7 and 1/9 are not currently supported, because the internal resolution can't represent them as integral numbers. This may change in a future version of Synfire.

The length 1/1 is equivalent to 4/4. It always refers to four quarter notes regardless of time signature, e.g. in 3:4 time, it exceeds one measure by a quarter note. This is where you would use 1m or 2m to denote a number of measures. The actual length depends on time signature currently in effect:

1m, 2m, 4m, 12m

Multiple expressions can be combined with + to break an odd length down into smaller units:

2m+2/4 4m+1/4\*

For especially odd lengths, a number of MIDI ticks can be appended to the expression. Synfire uses a clock resolution of 1920 ticks per 4/4 note.

8m+1/2+240

# Synchronization Settings For DAWs

### Settings known to work with several DAWs

Since Propellerhead's ReWire has been discontinued, <u>synchronizing Synfire with a DAW (on page 238)</u> requires slightly different settings per each DAW. Unlike ReWire however, current MIDI synchronization methods cannot alter a DAW's tempo. Therefore until we find a combination of sync hacks that might eventually fix this, you need to set the DAW tempo manually for now.

This information is maintained collaboratively here:

https://users.cognitone.com/tags/synchronization-daw

Related information Synchronizing With a DAW (on page 238)

# Feature Comparison

### What distinguishes the Sparks, Express and Pro editions of Synfire

Features of the **Sparks**, **Express** and **Pro** editions have been carefully balanced out in order to make every edition as useful as possible for their envisioned target audience and at the same time allow us to offer more affordable options as an alternative to the flagship **Pro** edition.

The **Sparks** edition is a great entry into the world of **Music Prototyping**. It helps you create and arrange songs in an exciting new way.

The **Express** edition offers more creative tools and more control over details. It is great entry point for home producers who don't yet need the advanced features of the **Pro** edition.

The **Pro** edition includes all prototyping features ever made for Synfire, offering the most comfortable and productive workflow for professionals and music enthusiasts.

	Synfire Sparks	Synfire Express	Synfire Pro
Target Audience	Songwriters, Bands, Teach- ers	Home Producers	Composers, Producers, Enthusiasts
Use Cases	Songwriting, building in- strumental parts for export to the DAW, teaching har- mony	Prototyping of arrange- ments, creating phrases with more detail and con- trol.	Composing orchestral mu- sic, advanced electronic music, game soundtracks, film scores, piano music, jazz, developing a personal style
	Harn	nony	
Interpretation	Presets	Voice Leading, Chromatic, Strategies	Voice Leading, Chromatic, Strategies, Voicing, Inver- sion, Alignment
Progressions	Embedded	Embedded, Standalone	Embedded, Standalone
Palettes	Embedded	Embedded, Standalone	Embedded, Standalone
Harmony Knowledge Base	Standard	Extended	Extended, Customizable
Harmonizer	√	$\checkmark$	$\checkmark$
Jazz Up/Down Progres- sions		$\checkmark$	$\checkmark$
Multiple Palette Windows		$\checkmark$	√
Catalog Editor		√	√
Merge Catalogs			$\checkmark$
Polytonality			$\checkmark$

### **Table 1. Synfire Editions**

### Arrangement

Structure	Song Parts	Nested Containers, Two Levels	Nested Containers, Unlim- ited Levels
Arrange Instruments	8	32	Unlimited
Sketch Instruments	8	24	24
Sketches	Create & Play	Fully Editable	Fully Editable

	Synfire Sparks	Synfire Express	Synfire Pro
Import	Single Phrases	Phrases, Libraries	Phrases, Libraries
Advanced Import Options		$\checkmark$	$\checkmark$
Time Signature Changes		$\checkmark$	$\checkmark$
Container Snapshots		$\checkmark$	$\checkmark$
Overview Page / Map			√
Continuous Tempo Changes			$\checkmark$
Cue Markers			√
Phrase Morphing			√
Inactivate Container, Inac- tivate Phrase (A/B Test- ing)			$\checkmark$

### Table 1. Synfire Editions (continued)

### **Phrase Editing**

Symbol Types	Horizontal, Chords, Bass, Pitch	Horizontal, Vertical, Chords, Bass, Pitch	Horizontal, Vertical, Chords, Bass, Pitch, Rela- tive
Parameter Interpolation		Use-Only	Recent, Linear, Snap, Dithered, Sequential, Hits
Extract From Figure, Out- put, Save To Figure		$\checkmark$	$\checkmark$
Inversion, Voicing, Bypass, Chromatic (Per Segment)		$\checkmark$	$\checkmark$
Advanced Figure Recogni- tion		$\checkmark$	$\checkmark$
Simulate Strumming, Ped- al, Broken Chords		$\checkmark$	$\checkmark$
Humanize, Enhance Beat		$\checkmark$	$\checkmark$
Shape Tool, Mute Tool		$\checkmark$	$\checkmark$
Focus On Symbol Type		$\checkmark$	$\checkmark$
Click Board		√	√

# Synfire SparksSynfire ExpressSynfire ProScale Board✓Text Data Entry✓Custom Quantize Grid✓Triplets, Quintuplets✓Split, Merge✓

### Table 1. Synfire Editions (continued)

### Parameters

Parameter Templates	Harmony	$\checkmark$	$\checkmark$
Harmony, Figure, Interpre- tation, Pan, Pause, Take, Volume	$\checkmark$	$\checkmark$	$\checkmark$
Custom Controllers		√	$\checkmark$
Bend, Breath, Expression, Length, Modulation, Out- put, Portamento, Preview, Reverb, Rhythm, Scheme, Shift, Step, Sustain, Trans- pose, Variation		√	√
Chromatic, Dynamics, Flow, Layer, Lyrics, Morph- ing, Skip, Tempo, Time			√

Library

Drag/Drop, Copy/Paste	$\checkmark$	$\checkmark$	√
Immediate Phrase Editing		$\checkmark$	$\checkmark$
Standalone Apps		$\checkmark$	$\checkmark$
Batch Import		√	√
Libraries With Sounds (Racks)		$\checkmark$	√
Generating Phrases		Variations of existing fac- tories	$\checkmark$
Preparation For Archive/ Sharing			√

### Table 1. Synfire Editions (continued)

	Synfire Sparks	Synfire Express	Synfire Pro
	Snip	pets	
Single-Instrument Groups		$\checkmark$	$\checkmark$
Mixed Groups (Folders)			$\checkmark$
Conversion of Arrange- ment to Snippets, Session to Arrangement			$\checkmark$
	Facto	ories	
Generate Random Phrases		Variations of existing fac- tories	$\checkmark$
Edit New Factories			$\checkmark$
	Audio	/MIDI	
Embedded GM Synth	$\checkmark$	$\checkmark$	$\checkmark$
External MIDI Ports	$\checkmark$	$\checkmark$	$\checkmark$
Plug-In Hosting	VST, VST3, AudioUnits	VST, VST3, AudioUnits	VST, VST3, AudioUnits
DAW Drones		VST, VST3, AudioUnits, AAX	VST, VST3, AudioUnits, AAX
DAW Drones (MIDI Ef- fects)			VST, VST3, AudioUnits, AAX
Record Parameters		$\checkmark$	$\checkmark$
Online Repository		$\checkmark$	$\checkmark$
Transport Remote Control		$\checkmark$	√
Sound Allocation Monitor		$\checkmark$	$\checkmark$
Custom Controllers		$\checkmark$	√
External Synchronization		Basic	SPP (MIDI Clock), MTC (MIDI Time Code), ReWire, OSC
Custom Synch Configura- tion			√
Custom Articulations			√
Device/Sound Templates			$\checkmark$

	Synfire Sparks	Synfire Express	Synfire Pro
Tagging Of Takes			$\checkmark$
Multiple Audio Engines			$\checkmark$
Export			
Printing	Tablature, Scales, Palettes	Tablature, Scales, Palettes	Tablature, Scales, Palettes
Drag & Drop MIDI	√	$\checkmark$	√
Standard MIDI Files	√	$\checkmark$	√
Transfer To Drone		$\checkmark$	$\checkmark$
Notation Formats			MusicXML, LilyPond

### Table 1. Synfire Editions (continued)

### **Upgrading To Express**

Upgrading from **Sparks** to **Express** much extends your possibilities to create, collect and edit phrases and gives you more leeway to arrange them in creative ways. Snippets allow for playing phrases as live loops.

### Arrangement

- Build more elaborate song structures with <u>containers (on page 21)</u> inside containers, nested up to two levels deep. Reuse containers in multiple places with <u>aliases (on page 22)</u>. Use containers that can overlap each other.
- Time signature changes in any container.
- Room for more instruments.

### **Phrase Editing & Parameters**

- Build more expressive Figures with an additional <u>Vertical (on page 295)</u> symbol type.
- Additional parameters to control rhythm independently of Figure: Step, Rhythm, Shift, Length, Velocity.
- Additional parameters to transform melodies and harmony: Transpose, Preview, Variation.
- MIDI controllers: Bend, Breath, Expression, Modulation, Reverb.
- Look at the <u>Output (on page 305)</u> parameter to see in an instant what Synfire is rendering from your input.
- Save and recall Parameter Templates for many parameters.
- Use the intuitive <u>Click Board (on page 91)</u> to transform a phrase and its parameters more comfortably.
- Use the **Shape** tool to warp parameter data.
- Use the Mute tool to silence individual Figure symbols.
- Focus on a single symbol type at a time for easier selection and editing.

### Library

- Edit Phrase Pools and phrases in a Library down to the parameter level. Do so inside the Arrange window or with a standalone <u>Library App (on page 149)</u>. Open multiple such apps at the same time to copy or move phrases with drag & drop.
- Create libraries with their own <u>Library Rack (on page 28)</u>, so you can collect phrases together with their original sounds. Drop phrases along with their sounds to new arrangements.
- Import MIDI files (on page 264) into a Library to collect new phrases for your songs.
- Edit your Sketches with the standalone Sketch App (on page 147).
- Generate any number of variations to existing factories.

### **Snippets**

• Place phrases in a Snippets (on page 24) grid and conduct them live and in real-time.

### Harmony

- Customize the <u>Catalog (on page 16)</u>, which offers a more extensive collection of chords and scales from the start. Leverage an extended harmony knowledge base.
- Keep multiple palette windows open at the same time to explore key changes (on page 10).
- Obtain more control over the <u>Interpretation (on page 301)</u> parameter with different voice leading strategies, filter settings and response times.
- Assign chord inversion and voicing to individual figure segments.

### Audio/MIDI

- Use **Drones** (on page 43) to host audio plug-ins for your instruments in a DAW and synchronize them with Synfire.
- Define your own Custom Controllers (on page 199).
- Record individual parameters (on page 261) directly.
- Access our Online Repository to share device descriptions with the user community.

### **Upgrading to Pro**

Coming from **Express**, an upgrade to **Pro** dramatically increases your creative freedom, workflow productivity and fun. Generate phrases to transform and experiment with them. Enjoy the power of **Music Prototyping** to the fullest. Additional sophisticated tools allow for more detailed control and help you develop your own personal style of music.

### Arrangement

- Build container structures with unlimited depth and a virtually unlimited number of instruments.
- Inactivate individual phrases and/or containers for A/B testing and experimentation.
- Navigate the <u>Overview (on page 100)</u> page to see the how your container hierarchy actually rolls out onto tracks, like in a DAW.

### **Phrase Editing & Parameters**

- Build more expressive Figures with an additional **<u>Relative (on page 296)</u>** symbol type.
- Define a custom <u>Grid (on page 88)</u> to make everything you draw or move around snap to arbitrary rhythmical positions. Drop phrases on the Grid outlet to use their rhythm.
- Additional parameters to control rhythm, dynamics and articulation: Skip, Flow, Dynamics.
- Additional parameters to transform harmony: Layer, Chromatic.
- Use <u>Morphing (on page 305)</u> to smoothly blend parameters, phrases and even containers into one another.
- Synchronize music to film using the Time parameter. Scale the tempo of your arrangement in order to match the beat with cue markers in a film.
- Use continuous Tempo (on page 313) changes anywhere.
- Assign an interpolation (on page 245) mode to parameters in order to achieve dramatic effects.
- Have triplets and quintuplets at hand from the toolbar to build more fluid rhythms.
- Freeze, split and merge phrases.
- Enter parameter data as text.

### Library

- Create libraries with their own <u>Library Rack (on page 28)</u>, so you can collect phrases, containers and snippets together with their original sounds. Drop phrases along with their sounds to new arrangements.
- Populate libraries with automatically generated phrases (see: Factories).
- Package finished libraries for archiving and sharing. Apply data reduction, time stamping, cleanup and more.

### **Snippets**

- Unleash the power of an advanced <u>Snippets (on page 24)</u> grid with Mixed Groups to compose, improvise and arrange music live and in real-time. Combine any number of parameters and phrases in a snippet to affect a session in real-time.
- Compute a snippet grid from any arrangement automatically.
- · Convert your last snippets session to a new arrangement automatically.

### **Factories**

- Configure your own custom <u>Factories (on page 29)</u> to generate random phrases that sound more musically sound than anything ever.
- The saying goes: Music is a language. Experience first hand how the power of a rules-based Artificial Intelligence Programming Language (KIM) turns this principle into natural sounding music fragments like nothing else.
- Generate thousands of unique phrases and variations. Enjoy a virtually endless supply of new musical expressions and ideas.

### Harmony

- Merge multiple catalogs (on page 16) and control what to do with conflicting or duplicate entries.
- Use polytonality (on page 15) to create multiple layers of Harmony.
- Obtain even more fine-grain control over the Interpretation (on page 301) parameter.

### Audio/MIDI

- Use MIDI Effects Drones (on page 44) to send rendered music to anywhere inside your DAW.
- Use <u>Custom Articulations (on page 201)</u> to control the timbres and playing styles of orchestral instruments and other big sound libraries.
- Run multiple Audio Engines (on page 41) in your LAN.

### Export

- Export MusicXML and LilyPond notation formats.
- Publish beautiful lead sheets using the Lyrics (on page 304) parameter.

Note:

The details of this comparison subject to change without notice.

# Glossary

### Terms not explained elsewhere but useful to know

Summary of terms in alphabetical order.

### Equivalence

Two chords or scales in **Twelve Tone Equal Temperament** that share the same pitch classes are said to be equivalent. This holds true for the chords Dm7(b5) and Fm6 or the scales C.natural-minor and G.phrygian, for instance. Equivalent chords or scales are not necessarily identical.

### Root

The first note of a scale or chord that also starts its name.

### Interval

The distance between two pitches measured in halftones.

### **Interval Structure**

The pattern of intervals that identify a chord or scale, independent of a root pitch. The interval structure of Am9 is thus m9, and the structure of F.harmonic-minor is harmonic minor.

### Rendering

The process of mapping all parameters to MIDI output.

### Texture

The overall timbre and rhythmical experience evoked by multiple instruments interacting with each other.

### **Twelve Tone Equal Temperament**

Also abbreviated as 12TET. Divides an octave into 12 equal halftone steps such that all 24 keys in the **Circle Of Fifths** share same pitches (frequencies) over all octaves. This allows for compositions to be transposed freely. Without 12TET, instruments need to be retuned for each key.

# Chapter 9. Boilerplate

Legal fine print and formal stuff

# Software License

Your right to use the software mentioned in this manual is governed by the accompanying software license agreement. The complete wording of this agreement can be found as a printed supplement, or along with the installation files you obtained from Cognitone, or under the Help menu of the software (*Help*). You are bound to this license agreement as soon as you install the software. If you do not agree with this license agreement, do not install the software. The purchase of a license grants you the right to use the software for personal or business purposes. However, you do not acquire ownership of the software.

# Copyright

© 2022 Cognitone GmbH. All rights reserved.

This document is protected by copyright law. As a legitimate owner of a license, you may print one copy for your personal use. Under the copyright laws, this manual may not be reproduced or copied, in whole or in part, or distributed in digital form, without the express written consent of Cognitone.

Cognitone, the Cognitone logo, Synfire, Music Prototyping Studio, Harmony Navigator, and other names are trademarks or registered trademarks of Cognitone in the European Union and other countries.

VST is a trademark of Steinberg Media Technologies GmbH. AudioUnits is a trademark of Apple Computer, Inc. ReWire is a trademark of Propellerhead Software. Other company and product names mentioned in this manual are trademarks of their respective companies and owners. Mention of third-party products is for informational purposes only. It does not constitute an endorsement or recommendation. Cognitone assumes no responsibility for these products.