zynaptiq

PITCHMAP

User Manual V1.5.0

zynaptiq

PITCHMAP

PITCHMAP

Introducing PITCHMAP

What is PITCHMAP?

PITCHMAP is the world's first and only real-time, polyphonic pitch correction and pitch mapping plug-in. Need to correct pitch inaccuracies on one instrument in a full mix or a polyphonic recording? Want to change the key of an entire song and simultaneously go from minor to major 7/9 on the fly? PITCHMAP allows you to do all of that, and more. To accommodate for modern musical genres and to open up a wide array of sound-design options, PITCHMAP also provides means of controllably making the process sound more synthetic in a very unique and evocative way.
Based on our proprietary **MAP** (**M**ixed-Signal **A**udio **P**rocessing) technology, PITCHMAP does all that by separating a musical signal into individual elements/sounds, including their associated harmonics and transients. Sounds to be processed are selected by their fundamental pitch, and their tuning can then be corrected or their pitches arbitrarily mapped individually, using pitch maps the user creates from within the GUI, or real-time MIDI data. On the fly and in unprecedented fidelity. The immediate nature of this process opens up new ways to intuitively interact with the compositional aspect of a recording, and unlocks a huge creative potential.

We sincerely hope you will enjoy using PITCHMAP as much as we did creating it. It is the culmination of 40 combined years of research and professional audio production expertise, and thus we are very excited to provide you with this unique and exceptional tool.

Now go create great audio!

Yours,
Stephan M. Bernsee & Denis H. Gökdag

zynaptiq headquarters
Hannover, Germany
September 2013

System Requirements

Mac Requirements

- Intel-based Apple Mac computer
- Minimum of 2 CPU cores running at 2.4 gHz or faster
- 1 GB of available RAM
- Mac OS X 10.6 or newer
- 60 MB free Hard-Disk space
- Apple AU (AudioUnits), Steinberg VST 2.4 or Avid RTAS/AAX compatible host software
- AAX requires Pro Tools version 10.3.5 or newer
- Internet Connection for Activation (though not necessarily on the Computer used for audio)

Windows Requirements

- Minimum of 2 CPU cores running at 2.4 gHz or faster
- 1 GB of available RAM
- Microsoft Windows XP SP3, Windows Vista or Windows 7/8
- 60 MB free Hard-Disk space
- Steinberg VST 2.4 or Avid RTAS/AAX compatible host software
- AAX requires Pro Tools 10.3.5 or newer
- Internet Connection for Activation (though not necessarily on the Computer used for audio)

Installation & Authorization

Installation Mac

Installing and authorizing **PITCHMAP** on a Mac is very straight-forward. Simply expand the ZIP file you received, navigate to the *Mac OS X* folder in the resulting directory and mount the contained disk image file (.dmg) by double-clicking it in the Finder. Then launch the installer contained on the resulting, auto-mounted volume. Follow the on-screen instructions to install.

The installer will install the following files to the following locations:

1) AudioUnits mono/stereo plug-in for both 32- and 64-bit use to <root drive>/Library/Audio/Plug-ins/Components/
2) VST mono/stereo plug-in for both 32- and 64-bit use to <root drive>/Library/Audio/Plug-ins/VST/
3) RTAS mono/stereo plug-in to /Library/Application Support/Digidesign/Plug-Ins/
4) AAX mono/stereo plug-in for both 32- and 64bit use to /Library/Application Support/Avid/Audio/Plug-Ins
5) Manual, PITCHMAP Authorizer app & stand-alone application to /Applications/Zynaptiq Plug-In Support/

During installation you will be prompted to activate your plug-in, so please keep your activation code at hand. Please note that we strongly advise to activate your software at this point, and not from within your host application.

Installation Windows

Installing and authorizing **PITCHMAP** on a Windows system is very straight-forward. Simply expand the ZIP file you received and navigate to the *Windows* folder in the resulting directory. Therein, you will find the following individual installers:

1) VST 2.4 mono/stereo plug-in (32bit)

1) VST 2.4 mono/stereo plug-in (64bit)

2) RTAS mono/stereo plug-in

3) AAX mono/stereo plug-in (32bit)

4) AAX mono/stereo plug-in (64bit)

Choose the appropriate installer and launch it. Please note that you need to be logged in as admin level user!

The installers create the following files:

1) The VST installer creates the PITCHMAP VST plug-in in <Program Files Folder>\Steinberg \VstPlugins. **Note:** should you keep your VST plug-ins in a different location, you should point the installer to that directory when provided with the option. **Note:** please do not rename, move or delete the "Data" folders installed next to the plug-ins

2) The RTAS installer creates the PITCHMAP RTAS plug-in as well as the PITCHMAP RTAS data folder in <Program Files Folder>\Common Files\Digidesign\DAE\Plug-Ins. **Note:** please do not rename, move or delete the "Data" folder installed next to the plug-in.

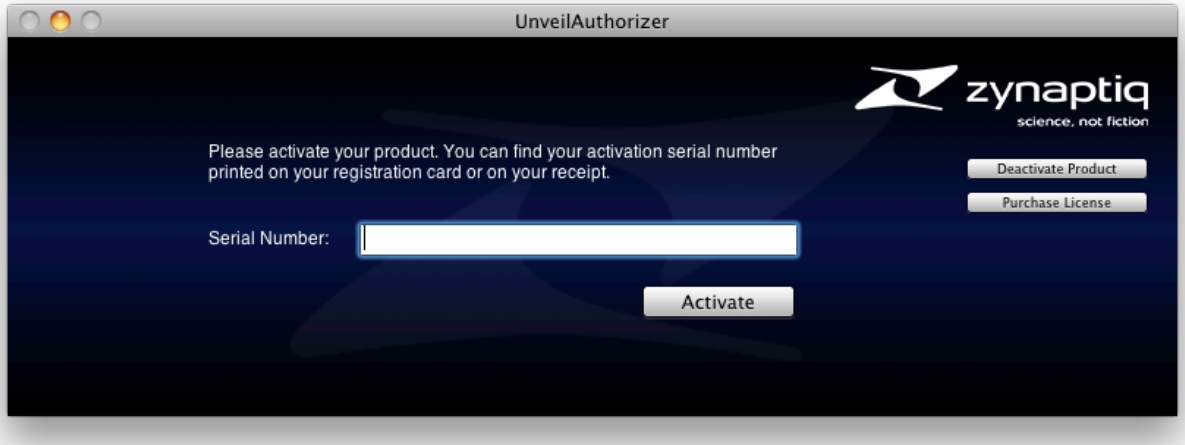
3) The AAX installers create the PITCHMAP AAX plug-in in <Program Files Folder>\Common Files\Avid\Audio\Plug-Ins.

All Windows installers will install the PITCHMAP Authorizer app to <Program Files Folder>\Zynaptiq\PITCHMAP\.

Windows Vista, Windows 7/8 users: please make sure you activate the plug-in when the activation dialog is opened during install (and NOT from within your host!).

Authorization Mac & Windows

To be able to use your new software, it needs to be activated. During installation of **PITCHMAP**, the **PitchmapAuthorizer** app will automatically be launched.



Enter your serial number and click “Activate”. If your machine has access to the internet, you’ll be shown a confirmation of the successful authorization and you’re good to go. If the machine you’re activating on has no internet connection, you will be guided through the process.

Authorization Details

Generally, our copy protection mechanism will allow you to have two machines authorized at the same time. It is essentially a challenge-response type system that takes specifics of your computer into account anonymously. However, unlike most other challenge-response systems,

the Zynaptiq copy protection allows the user to de-activate a system at any time, and re-activate it later - without contacting us. So essentially, you can pretty much move your authorizations around as needed. The only limitation is that in order to de-activate a machine, it needs to be connected to the internet.

De-Activating a Machine

To free up an activation, you can de-activate a machine. To do this, please launch the **PitchmapAuthorizer** app, and click “Deactivate Product”.

You will be presented with a confirmation message and you’re done.

Please note that the machine to be de-activated needs to be connected to the internet!

Overview

Features at a Glance

PITCHMAP provides all of the following functionality in real-time:

- Automatic correction of pitch inaccuracies in polyphonic and mixed signals
- Mapping of source to destination pitches via the GUI
- Definition of destination pitch grid via live MIDI
- Process bypass on arbitrary notes selected by their fundamental pitch
- all of which leaves non-pitched signals like drums virtually untouched
- Suppression & Extraction of mix elements based on their pitch using Mute Filters via GUI or MIDI control
- Continuously variable unique synthetic coloration options via parameters Electrify and Purify
- Polyphonic portamento/glides using the Glide parameter
- High-speed workflow: options to rapidly generate complex pitch maps using macro controls
- Snapshot functionality to allow rapid creation of automation
- Mono/Stereo plug-in supporting 44.1kHz and 48kHz sampling rates (higher rates being worked on)

Applications And Usage Scenarios

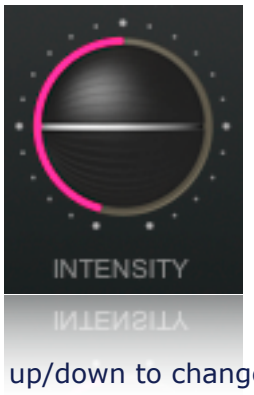
PITCHMAP has quite a lot of uses. Here are a couple of them.

- Fixing tuning inaccuracies in mixed signals
- Adapting one recording, song, loop or sample to fit another
- Playing own compositions using a recorded song or loop as an “instrument” using live MIDI input to map pitches
- Changing the key/scale of a recording/song/loop/sample
- Changing the pitch of individual notes within a recording
- Suppressing mix elements
- Extracting/Isolating mix elements
- Rapid song prototyping
- Creative sound design
- Speeding up sample-based music production, re-mixing and mash-up creation workflows significantly

The Parameters

General Controls Behavior

PITCHMAP sports several control types, which have a variety of usability features. We will describe the ones with extra functionality here.



"Trackball" Slider

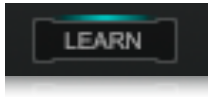
This control is a unique Zynaptiq type - it is essentially a vertical fader, that looks like a knob. The benefits of this configuration are that they are simple to control with a mouse like a fader, yet conserve GUI space like a rotary control. Also, the combination of the horizontal marking on the "Trackball" and the circular value display make it very easy to tell the current value of the parameter.

- Drag up/down to change the parameter value.
- Hold shift while dragging for finer resolution.
- When the control has focus, the up/down arrow keys and the mouse scroll-wheel can be used to increment or decrement the value. The control has focus whenever the mouse hovers over it.
- CTRL-click or right-click on the control to reset it to default.
- Double-click on the control opens a text box for numerical entry of the value.

“Standard” Slider

This is a straight-forward slider control, which comes in horizontal as well as vertical configurations. Essentially, it follows the same rules as the “Trackball”:

- Drag the thumb along the slider axis to adjust the value
- Click anywhere in the slider groove to set the value directly
- Hold shift while dragging for finer resolution.
- When the control has focus, the up/down arrow keys and mouse scroll-wheel can be used to increment or decrement the value. This control type has focus after it was clicked on AND the mouse hovers over it.
- CTRL-click or right-click on the control to reset it to default.
- Double-click on the control opens a text box for numerical entry of the value.



Toggle Switch

A latching switch that activates a function or switches between two different functions. CTRL-click resets it to default.

Momentary Button

A momentary switch that triggers an action, such as a reset/clear. This control type comes in several sizes, but is always essentially a rectangular outline.

Pull-down Selector/Menu

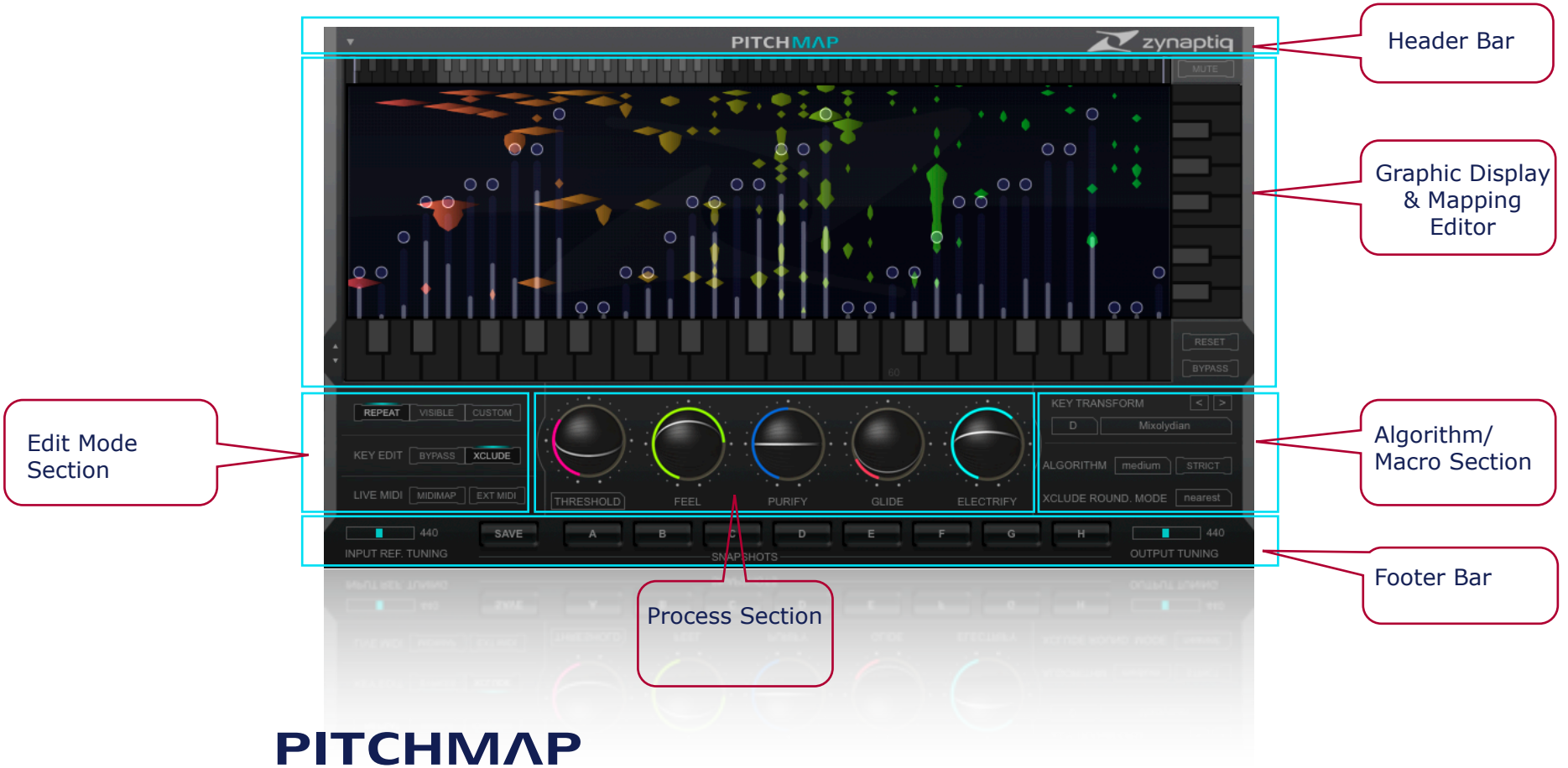
Opens a menu on click. You can recognize this control type by its “cut off” top right corner. CTRL-click resets this to default value.

Checkbox Switch

Functionally identical to the Toggle Switch, this switch type is used within the display area.

General GUI Layout

The **PITCHMAP** user interface is divided into 6 sections: the Header- and Footer Bars, the Graphic Display & Mapping Editor Area and the Edit Mode, Process and Algorithm/Macro Sections.



The Header Bar

The Preset Manager

To make working with the different plug-in formats easier, PITCHMAP provides it's own preset manager and preset format. This way, you can load your custom settings exactly the same way regardless of whether you're working on the Mac or on a Windows machine, whether you're using RTAS or VST.

The Preset manager is accessed using the small triangle icon in the upper left corner of the GUI:

A click on the triangle icon brings up a menu that provides options for saving and loading presets, saving the current setting as default and re-setting the default preset to factory default values. If you edit a recalled preset, the triangle icon will be lit up to indicate that the current and saved values don't match.

Storage Locations

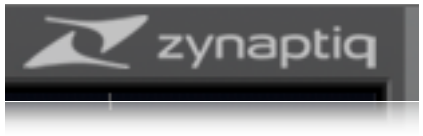
There are separate locations for factory presets and user presets, and they both depend on the OS platform you're working on:

Factory presets (Mac OS X): these are stored inside the actual plug-in.

Factory presets (Windows): these are stored in the PITCHMAP VST/RTAS/AAX Data folder, which is installed next to the plug-in.

User presets (Mac OS X and Windows): these are stored to <your user document directory>/Zynaptiq/PITCHMAP/Presets/

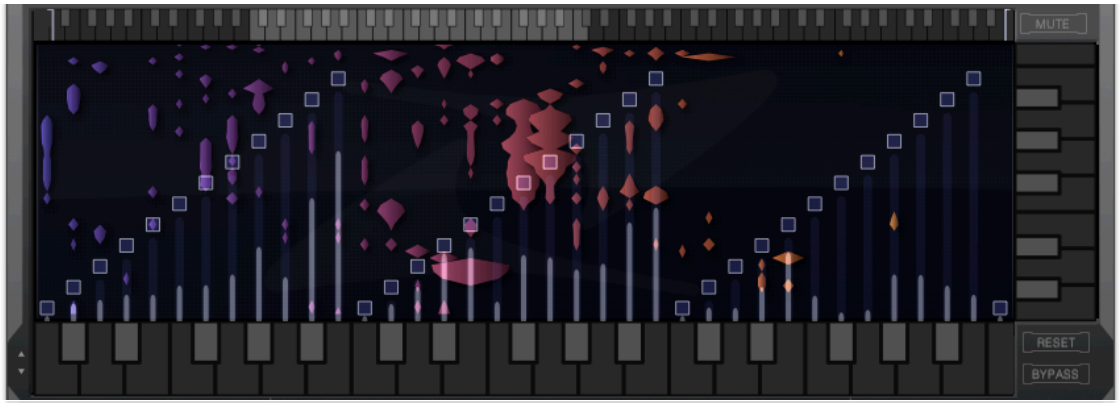
“About” Screen & Update Check



Clicking on the Zynaptiq Logo opens an “About…” screen which displays version information as well as copyright and and legal notices. Also, from within the About screen, you may check for updates manually or toggle automatic weekly update checking.

The Graphic Display & Mapping Editor

Overview



Display

The *Display* is your central source of information about your signal in PITCHMAP. Understanding it and the functionality it provides is essential to getting the most out of PITCHMAP. Basically, it displays 3 octaves of your input signal in a fashion similar to a spectrogram, flowing from bottom to top, with lower frequencies to the left and higher frequencies to the right. However, it is not a simple spectrogram that maps frequency content and amplitude to horizontal position and color, it is significantly more advanced than that. What we display are detected sounds, including their harmonics, transients and noise components, whose fundamental pitch we map horizontally. Absolute pitch is coded into the color, and amplitude is displayed using the width of the symbols. The *Lower Keyboard* at the bottom of the *Display* also indicates their pitch, serving as legend or grid. In a way the *Display* is rather similar to a piano roll.

The *Display* also serves as background for the *Pitch Mapping Sliders*, helping you find the pitches you want to process.

The Upper Keyboard Thumbnail

The *Upper Keyboard Thumbnail* is used as reference grid for the *Low-Cut* and *High-Cut* sliders, and for navigating the *Display*. The lighter grey 3-octave area corresponds to the visible 3 *Display* & *Lower Keyboard* octaves. Drag the grey area or click anywhere in the upper keyboard to scroll the *Display* and the *Lower Keyboard*.

Mute Filters: Low-Cut And High-Cut Sliders

Sounds outside the bounds defined by these sliders are either bypassed or muted, depending on the state of the *Mute* switch. Drag these to adjust the cut-off point. CTRL-Click anywhere on the *Upper Keyboard Thumbnail* to reset these. When the *Low-Cut* slider is dragged to the right past the *High-Cut* slider, the range between the sliders is bypassed/muted, not the range outside of the sliders. Please note: THESE ARE NOT FILTERS! When in *Mute* mode, any sound that has it's base pitch in the area that is *Mute*d will be removed, including its harmonics, transients and noise components. Think in terms of muting channels on a mixing desk.

Mute Switch

This control defines whether the material outside of the boundaries defined by the *Low-Cut* and *High-Cut* parameters is *Bypass*ed or *Mute*d.

Lower Keyboard

The *Lower Keyboard* has multiple functions. Firstly, it serves as a reference grid for the sounds shown in the *Display*. Secondly, it serves as source pitch grid for the *Pitch Mapping Sliders*. Thirdly, it is used to edit the *Bypass* and *Xclude* states per source pitch. Think of the keys of the *Lower Keyboard* like mixer channels (with every pitch that is contained in your recording having its own mixer channel) and you'll get the hang of it pretty fast. Also displays which MIDI note is being pressed when in *MIDI MAP* mode.

Behavior influenced by:

A click on a key toggles the *Bypass/Xclude* state for that pitch, depending on whether *Bypass* or *Xclude* is selected in the *Key Edit* parameter. A *Bypass*ed key will show green, an *Xclude*d key orange/red.

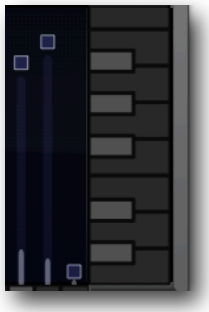
Also, *Repeat/Visible/Custom* is respected. So, for example, if *Repeat* is on, a click on a key will toggle the state for that key in all octaves. CTRL-Clicking anywhere on the *Lower Keyboard* resets all states. When in *MIDI MAP* mode, the *Lower Keyboard* serves display purposes only, and keys pressed via MIDI will be shown in blue.

Finally, a SHIFT-Click on one of the keys will play a sine wave at that pitch.

Reset

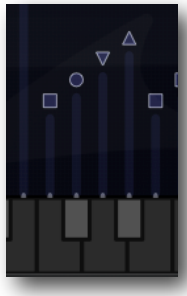
Resets all *Pitch Mapping Sliders* to default values. Use this to start from scratch. Alternately, you can CTRL-click anywhere in the *Display* to reset the sliders.

Right Keyboard



The *Right Keyboard* represents the destination (output) pitches and serves as a grid for the *Pitch Mapping Sliders*. The keys light up when adjusting a slider to show its value.

Pitch Mapping Sliders



These are the main GUI controls for mapping pitch. By dragging these up or down, you are mapping a source pitch (as shown by the *Lower Keyboard*) to a destination pitch (as shown by the *Right Keyboard*). Think of this like a routing matrix or patchbay, "this goes there". Or you can think of a slider as a transpose/pitch-shift value that is available per source pitch. While dragging a slider, the relevant pitch is soloed. Holding alt/option while dragging drags along the sliders of the same pitch class across all octaves (so dragging the slider of a C drags the sliders every C). The body of the slider shows a level meter for the pitch that is referenced. The head of the slider adjusts the mapping behavior, and can be switched through 4 states by shift-clicking it.

Square Slider Head

Pitch is mapped within the octave of the source pitch only, exactly defined by setting a slider. WYSIWYG.

Round Slider Head

Mapping is automatically performed towards the nearest octave. For example: when you're mapping a C to an E, the E in the same octave as your source pitch will be used (it has a distance of +4 semitones to the C). If you mapped the C to an A (+9 semitones), the lower octave will be used (-3 semitones). The purpose of this mode is to keep transposition as low as possible to maximize sound quality, and it is actually quite similar to the way musicians voice their chords intuitively.

Downward Triangle Slider Head

Mapping always uses the octave below, effectively transposing your slider down by an octave. The purpose of this mode is to allow transpositions outside of the range visualized by the *Right Keyboard*, while keeping it easy to read the map.

Upward triangle Slider Head

Performs the same thing as the *Downward Triangle Slider Head*, but transposes *up* by an octave instead of *down*.

Bypass

Performs a highly complex function best described as *Bypass*ing the plugin.

Edit Mode Section

Edit Mode

The active *Edit Mode* defines behavior when editing *Pitch Mapping Sliders*, *Bypass/Xclude* states via the *Lower Keyboard* and when using *MIDI MAP*.

Edit Mode: Repeat

When *Edit Mode* is set to *Repeat*, any value edited is copied to all octaves. Example: when dragging a *Pitch Mapping Slider* associated with a "C", the same value is applied to every "C" in all octaves. This mode is useful for quickly mapping one pitch class to another across the entire spectrum. When using *MIDI MAP*, playing a chord results in the entire range being mapped to that harmony, giving results instantly.

Edit Mode: Visible

Visible works like *Repeat*, but restricts the range to the visible three octaves. Example: setting *Bypass* enabled using the *Lower Keyboard* for an "A" causes all "A" keys in the visible three octave range to be set to the same value. *Visible* mode helps tailor independent intra-scale voicings for different frequency ranges (basically for bass, harmonies and melodies). In *MIDI MAP* mode, the range outside of the visible area is divided into two separate *Custom* zones.

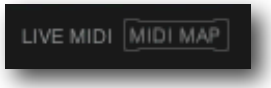
Edit Mode: Custom

In the *Custom* setting, only the exact key/slider/note that you edit is changed, leaving all other values alone. In *MIDI MAP* mode, this allows playing completely independent phrases in varying parts of the MIDI keyboard. This even allows mapping all source pitches to a single destination pitch (which can sound very cool)!

Key Edit Mode

The active *Key Edit Mode* selects whether clicking on a key on the *Lower Keyboard* edits the *Bypass* or *Xclude* state of the corresponding pitch. When in *MIDI MAP* mode, selects whether MIDI notes are used to map the pitches (*Xclude*) or to un-mute a pitch (*Bypass*)

Live MIDI: MIDI MAP



MIDI MAP is a unique PITCHMAP feature that allows forcing melodies or harmonies you play via MIDI onto the signal being processed. It's like you were playing the instruments contained in the signal via MIDI, all at the same time. *MIDI MAP* also allows intuitive de-mixing by allowing only sounds to pass through that correspond to the MIDI notes you are playing. When *MIDI MAP* is active, the *Lower Keyboard*, including all associated *Bypass* and *Xclude* states, is ignored. Instead, live MIDI input is used to set values. Depending on the state of the *Key Edit* parameter, MIDI notes either define the destination pitches allowed (*Xclude*) or are used to un-mute a key (*Bypass*). *MIDI MAP* respects the *Edit Modes Repeat/Visible/Custom* setting.

See the Quick-Start Tutorials on how to set up MIDI with PITCHMAP in your host application.

EXT. MIDI

This feature is specific to the AudioUnits version of the plug-in. If your host does not support sending MIDI to plug-ins, you can activate *EXT MIDI*. This causes any MIDI signal sent to your DAW to be "heard" and used by PITCHMAP. If your host DOES support sending MIDI to plug-ins, please do not activate this!

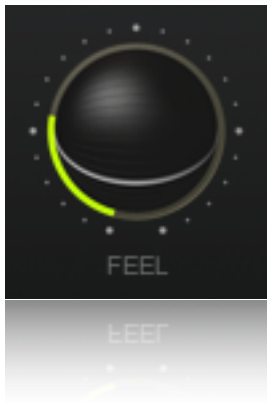
Process Section

THRESHOLD

Allows automatically *Bypass*ing notes that are detuned less than an amount set with this control. At minimum *Threshold*, everything is processed. With higher values, notes that are already somewhat accurately tuned will be *Bypass*ed. For creative applications set this low, for fixing tuning issues in sensitive material set this high.

Clicking on the *Threshold* label toggles between *Threshold* and *Global Threshold*. The difference between the two is whether or not transposed notes will be bypassed (*Global*) or not.

FEEL



Allows re-introducing micro-variations in pitch, such as vibrato, after correction is applied. Set low for very tight or synthetic results as popular in many recent genres, medium for tightened and natural results, and high to preserve all intonation detail (while still having any pitch mapping applied).

PURIFY

Purify adjusts the amount of noisy components. Values higher than the default 50% reduce noisy components and introduce an effect reminiscent of resonance, values below 50% increase the the level of noisy components. Can be used to deliberately create a surreal sound or to bring focus to transients and other non-harmonic aspects of the signal. High values increase CPU load.

GLIDE

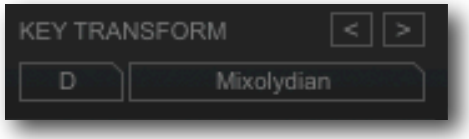
Glide adjusts the length of polyphonic glide/portamento to be applied. Whenever a new sound starts, the pitch ramps up/down from the source pitch to the destination pitch over an amount of time set with this slider. Subsequent sounds on the same pitch do not trigger the *Glide* again, unless interrupted by a non-pitched transient. Pretty cool.

ELECTRIFY

Electrify can be used both to introduce a very unique, electric synthetic coloration and to optimize the process to your signal. The default value is 50%, which works well for most situations. High values make results sound electric, low values can actually improve processing quality but may introduce unexpected harmonics when working with sparse recordings. Values around 60-75% can work well to improve results when working with mixed vocal stems. High values work best in combination with a low value for *Feel*. Note that setting *Electrify* to very low values increases the number of sounds detected and will thus use more CPU for the *Display*. Technically, this controls adjusts how many sounds are being tracked - at maximum position, only one sound is tracked.

Algorithm & Macro Section

KEY TRANSFORM Macros



Key Transform is a macro function that can be used to quickly create a *Pitch Map* based on a destination key/scale by setting all *Pitch Mapping Sliders* to specific values. *Key Transform* consists of two pull-down menus for selecting destination root key and destination scale. The macro is applied on releasing either of the pull-down menus. Also, two "voicing" buttons are provided. Clicking these will shift all *Pitch Mapping Sliders* left or right by one semitone per click. The result is that the key/scale stay the same, but the input pitches are rotated against the map - similar in concept to chord inversions or the "degree" within a scale.

As PITCHMAP does not make assumptions on the input key/scale, *Key Transform* does not transform relative to the input key/scale. Instead, it transforms relative to a C chromatic scale (which is what the *Pitch Map* defaults to). What does this mean in practice? It means that if your input scale is D minor, and you select A Major, input pitches will be mapped to output pitches in the following way:

Input	PITCHMAP	Classic
D	B	A
E	C#	B
F	D	C#
G	E	D
A	F#	E
A#	F#	F#
C	A	G#

As you can see, the results are rotated against what you'd expect to happen from the perspective of classic music theory. Also, you'll note that the G# is missing. To align the two tables, press the *Right Voicing Arrow* two times. Why two? Because that is the difference from C (which we base our transform on) to D (the scale the input has), in semitones.

This will result in:

Input	PITCHMAP	Classic
D	A	A
E	B	B
F	C#	C#
G	D	D
A	E	E
A#	E	F#
C	F#	G#

So we're getting close, but what's up with the E showing up twice and the G# missing again? Well, that's because we deliberately chose an example that highlights the caveats of our approach.

Key here is understanding that the notes in different scales are not equally spaced. Also, scales can have a different number of intervals - for example, classic western scales have 7 intervals, but Pentatonic scales have 5. Neither of the two is a divisor of 12 (the number of steps an octave is divided into). So there are bound to be rounding errors, and there really isn't a purely mathematical, "correct" method to decide whether an interval that exists in one scale should be mapped to the next higher or the next lower interval when going to a scale that doesn't have this interval. This choice would depend on user preference, direction of movement within the scale, and a whole bunch of other factors. Also, whether to transpose a source pitch that is in neither of the source or destination scales up or down is a decision that a musician would base on a similar set of considerations - a choice we do not want to make for you.

In a nut-shell: since PITCHMAP doesn't identify the source key/scale, we must generalize the transform, so that it works with *any* input signal, regardless what key/scale it is in. All this results in some compromise if you look at the process from a music theory point of view - while from a practical perspective, our approach gives good results in nearly all cases.

Well, why is it that we don't track what the input key/scale is? The answer is quite simply that it isn't really possible to do this in real-time, as there are ambiguities that can only be resolved by looking at the context the pitches are in. For example, the notes (E, G, B) may either be an E Minor... or a G Major that omits the fifth and adds a 6th. You can tell the difference only by looking at what was being played before and after this chord, and even then, there's quite

often room for interpretation. So in order to reliably discern the source key/scale, you'd really need to drop out of real-time and do some sort of learning/pre-analysis, that looks at the whole file or at least a longer section. We specifically did NOT want to go down that route, as PITCHMAP is all about immediate results and a musical workflow.

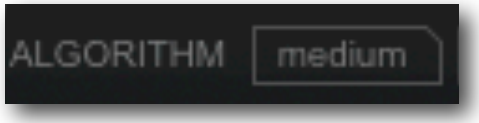
The good news is that all of this really doesn't matter very much in practice. To change the key/scale of an input to a desired target key/scale, all you really do is

- select the target key/scale using the *Key Transform* macro, then
- click on the left/right *Voicing Arrows* until you like the results (if you didn't already like them before), and
- fine-tune the map by adjusting 1-2 sliders manually if needed.

Of course there's nothing stopping you from manually mapping pitches from scratch if you're all set on a specific input-to-output map. Or you can choose to use *MIDI MAP* mode, which solves most of the issues described above - instead of mapping each possible pitch to a specified target pitch, *MIDI MAP* mode simply sets up a grid of allowed target notes and everything just slides into place.

Note that as *Key Transform* is just a means of setting the *Pitch Mapping Sliders*, it doesn't "remember" what key/scale you have selected earlier when closing and re-opening the UI, as you may have made changes to the sliders anyway.

ALGORITHM



Selects *Linear*, *Medium* or *Natural* algorithm modes. In *Medium* and *Natural* modes, the analysis engine uses a perceptive model to discern voice components in the input signal, and process these separately. This can prevent a certain type of coloration when processing signals that are musically rather "busy" and contain voice. If you notice unexpected coloration of voice components, try different values here. Also, for isolating sounds, try the different settings to see which *Algorithm* mode works best.

STRICT

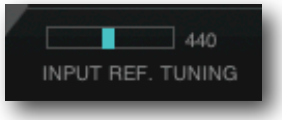
This switch toggles between normal and *Strict* pitch correction modes. *Strict* removes more pitch variation, but may reduce transient crispness.

XCLUDE ROUND. MODE

Selects the rounding mode used when *Xclud*ing notes. This influences the way that pitches that lie between allowed destination pitches are handled, e.g. in which direction they are moved under which circumstances. Available options are *Up*, *Down*, *Nearest* and *Intelligent*. *Intelligent* tries to avoid jumping as much as possible, *Nearest* creates typical (and quite popular) tuning-effects.

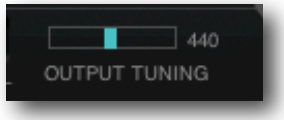
The Footer Bar

INPUT REF. TUNING



This slider allows adjusting the input/source reference tuning, e.g. what the algorithm sees as being “in tune”. This value is in Hertz, representing the absolute frequency of what is considered as an A4 (typically 440.00 Hertz). If your input signal is not tuned to 440Hz, you may see many “tuning jumps”. Adjusting this slider may reduce that.

OUTPUT TUNING



Adjusts the output tuning. Does not affect *Bypass*ed components. Use this if you want to have your results fit another recording that is not tuned to A=440Hz.

Snapshots

The *Snapshots* allow saving different pitch and key maps within one plug-in preset. Use this feature for example if different parts of a recording need different mapping/xclude/bypass settings. Using the *Snapshot* functionality, you can create one setting per song part and call these using host automation. Automation would otherwise be very time-consuming to say the least. *Snapshots* store the state of the *Pitch Mapping Sliders* and the states of the *Lower Keyboard* keys, but NOT the values of all other parameters. As these can be automated using your host application, storing these with the *Snapshots* would create potential conflicts.

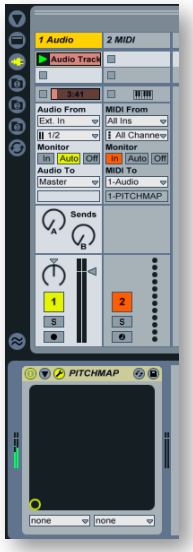
Snapshots are saved by alt/option-clicking on the slot you wish to save to and recalled by a click on the *Snapshot* slot to be recalled. Alternately, you can first click on the *Save* button, then on the *Snapshot* slot you wish to save to. To cancel, click *Save* again (and don't click on a *Snapshot* button). Both methods overwrite the targeted *Snapshot* slot without further warning, so be careful!

Tutorial A: MIDI Modes and Setting Up MIDI Control In Your Host

There are two modes of MIDI operation in PITCHMAP: *MIDI MAP* and "regular". The regular mode is always on, except if *MIDI MAP* has been activated. In regular mode, incoming MIDI notes will toggle the state of the corresponding key of the *Lower Keyboard*, respecting the *Xclude/Bypass* and *Repeat/Visible/Custom* settings. So pressing C3 on a MIDI keyboard while in *Xclude* and *Custom* modes will cause the C3 key of the *Lower Keyboard* to be activated if it was inactive before, and vice-versa. In *MIDI MAP* mode, incoming MIDI notes define either the grid of allowed target notes when in *Xclude* mode, allowing you to literally play the target harmonies and melodies via MIDI, or which notes to play back at all when in *Bypass* mode.

Ableton Live

To control PITCHMAP via MIDI in Ableton Live:



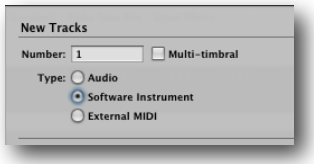
- You'll need one audio track and one MIDI track
- Insert the plug-in on the audio track to be processed
- Create a MIDI track
- Select your MIDI input device as input to the MIDI track, set monitoring to "In" and
- Select PITCHMAP as destination for the MIDI track

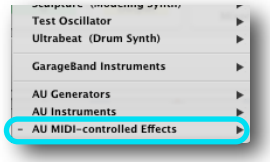
To use *MIDI MAP* mode to change the harmonies and melodies of your audio, simply load the preset *01 MIDI Map Template - Medium* located in the preset folder *02 MIDI Templates.*

Logic Pro

To control PITCHMAP via MIDI in Logic Pro:

- Insert the plug-in into an instrument track, you'll find it in the category "AU MIDI-controlled Effects"





- Place the audio file to be processed on an audio track, and set the track's output to "No Output".
- Choose the track that's playing back the audio as the side-chain source from within the PITCHMAP GUI.

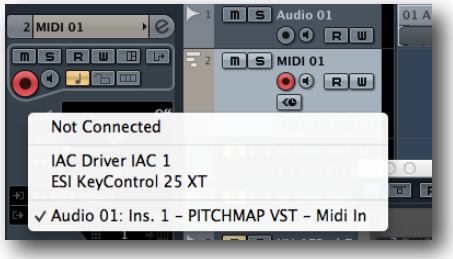
- Select the Instrument track in the Arrangement view, the red "R" button on the track header should be lit up.

To use *MIDI MAP* mode to change the harmonies and melodies of your audio, simply load the preset *01 MIDI Map Template - Medium* located in the preset folder *02 MIDI Templates.*

Cubase

To control PITCHMAP via MIDI in Cubase:

- Insert the plug-in on the audio track to be processed
- Create a MIDI track
- Select your MIDI input device as input to the MIDI track, activate monitoring and
- Select PITCHMAP as destination for the MIDI track



To use *MIDI MAP* mode to change the harmonies and melodies of your audio, simply load the preset *01 MIDI Map Template - Medium* located in the preset folder *02 MIDI Templates.*

Pro Tools

To control PITCHMAP via MIDI in Pro Tools:

- Insert the plug-in on the audio track to be processed
- Create a MIDI track
- Select your MIDI input device as input to the MIDI track and
- Select PITCHMAP as destination for the MIDI track

To use *MIDI MAP* mode to change the harmonies and melodies of your audio, simply load the preset *01 MIDI Map Template - Medium* located in the preset folder *02 MIDI Templates.*

Tutorial B: Correcting Tuning Issues in Mixed Signals

Here's a quick run-down on how to correct tuning issues in mixed signals.

- Insert PITCHMAP into an effects slot of the audio track that needs pitch correction.
- Load the preset *001 Tighter Pitch* from the preset folder *01 Tuning And Chord Changes*
- If you need stronger pitch correction, lower *Threshold* and *Feel* until all notes you wish to correct are being corrected.
- If you want less pitch correction, increasing *Feel* re-introduces the pitch deviation (unless you want a 100% tuned result, in which case you should set *Feel* to 0%), and increasing Threshold will bypass processing on all notes that are already close to being in tune.
- If you're not getting enough pitch correction even with *Threshold* and *Feel* at zero, which may happen in rare circumstances, try checking *Strict*.
- If your signal contains drums and they're adversely affected, try raising *Threshold* a little, decreasing *Purify* slightly, or using the *Low-Cut* and *High-Cut Sliders* at the top of the *Display* to *Bypass* very low and very high components (which typically represent kick drums and hi-hats/shakers).
- If there are parts of the signal that you wish to explicitly exclude form pitch correction, such as a vocal line with lots of performance detail, you may want to *Bypass* the relevant pitches. To do this, select the *Key Edit Bypass* mode and *Bypass* all notes you wish to remain unprocessed by clicking on the respective key of the *Lower Keyboard*. Note that the *Edit Modes Repeat/Visible/Custom* apply (see the relevant section of this manual). You will probably want to set *Edit Mode* to *Custom* for most cases covered by this tutorial. When *Bypass*ing a pitch, its *Lower Keyboard* key turns green, the sounds displayed in the *Display* are grayed out, and the associated *Pitch Mapping Slider* is hidden.

- If you are hearing a lot of "tuning artifact pitch-jumps", chances are that the input signal is not tuned to A=440Hz. Try adjusting the *Input Ref. Tuning* slider to see whether this helps.

Tutorial C: Attenuating Wrong Notes

Here's a quick run-down on how to selectively attenuate individual wrong notes in an otherwise good performance. Here we take care of notes that weren't supposed to be there *at all*, as opposed to notes being played in a wrong pitch accidentally.

- Insert PITCHMAP into an effects slot of the audio track that needs fixing.
- Load the preset *03 Suppress Pitched* from the preset folder *04 Mixing*
- Set *Edit Mode* to *Repeat* and *Key Edit* to *Bypass*.
- *Bypass* all keys by clicking on 12 successive keys of the *Lower Keyboard*; they'll turn green and you'll hear your unprocessed signal.
- Switch *Edit Mode* to *Custom* and mute the offending notes by removing their *Bypass* (which routes them to the *Mute Filters*, hence, they're attenuated).
- If "muted" sounds are fluttering between on and off, or if harmonics aren't all attenuated evenly, try adjusting the *Input Ref. Tuning* and switch between the different *Algorithm* modes to find the one that works best.
- Also, modifying *Purify* and *Electrify* can affect the amount and precision of suppression.

Tutorial D: Changing the Key/ Scale/Melody of a Recording using the GUI Macros.

Here's a quick run-down on how to change Key/Scale/Melody of a recording using a *Pitch Map* created using the *Key Transform* macros of the GUI as starting point.

- Insert PITCHMAP into an effects slot of the audio track that you want to process
- Load the preset *002 Much Tighter Pitch* from the preset folder *01 Tuning And Chord Changes* as a starting point.
- If you already have an idea what target key/scale you want, select the appropriate key and scale from the *Key Transform* pull-down menus.
- You'll probably hear pitch going all over the place. To remedy this, read on.
- Move all sliders to the left or to the right using the *Voicing Arrows*. While you've set the destination pitch grid using the *Key Transform* macro, in this step you define which notes go to which grid lines. Think of this as setting the chord inversions. There'll be settings that have significantly less pitch shifting - you can tell how much shifting is applied by hovering the mouse over the *Display* - the shift values per slider are displayed at the top of the *Display*. As a rule of thumb, shift left or right by the number of semitones the target root note differs from C, and use the shortest distance with regard to octaves - so if you are going for a D, click the right arrow twice, if you're going for an A, click the right arrow three times.
- Try changing the *Slider Heads* through their modes, to see whether any of the settings is preferable.
- Once you have found a setting that is close to what you're looking for, you can fine-tune the pitch mapping. You can either adjust individual sliders to route the sounds they reference to different target pitches, or you can remove specific pitches from the grid by using *Xclude*. To edit all sliders of one pitch class (for example, all C notes), use *Repeat* mode, and to edit just one particular slider, use *Custom* mode. You can *Xclude* keys by first selecting *Key Edit Xclude* mode and then clicking on the keys of the *Lower Keyboard* you wish to remove. The

key will turn orange, its slider is hidden, the sound associated is grayed out in the *Display* and forced to one of the neighboring, un-*Xcluded* pitches. Which direction it is forced to depends on the *Xclude Round. Mode*. Please note that editing *Xclude* states on the *Lower Keyboard* respects the setting *Repeat/Visible/Custom*...play with this to learn how the parameters interact.

- Finally, fine-tune the process to minimize unwanted artifacts. Raising the *Threshold* slightly may help preserving drums and other transient material better. Changing *Xclude Round. Mode* and adjusting *Input Ref. Pitch* can help minimize pitch jumping. Raising or lowering *Feel* can help getting coherent results, and *Purify* can be used to clean or dirty up the sound. Also, if there's a lot going on, you can loose some of the highest pitches by activating *Mute* and pulling down the *High-Cut* slider gradually.

Tutorial E: Changing the Key/ Scale/Melody of a Recording using the GUI Sliders

Here's a quick run-down on how to change Key/Scale/Melody of a recording manually using the *Pitch Mapping Sliders* of the GUI.

- Insert PITCHMAP into an effects slot of the audio track that you wish to process.
- Load the Default setting as starting point, and set *Threshold* and *Feel* to minimum.
- Drag the *Pitch Mapping Sliders* up or down. The sound associated with a slider is soloed while dragging, and the *Right Keyboard's* keys light up to show you what pitch you're mapping to.
- To adjust one slider at a time, select the *Custom Edit Mode*. To adjust all sliders associated to the same pitch class across all octaves, select the *Repeat* mode, to restrict that to the three visible octaves, select *Visible*.
- When in *Custom* mode, hold ALT/Option while dragging to temporarily enable *Repeat* mode.

- Slowly raise *Threshold* to the point where some notes become unprocessed, then back it down a little. This way, you're excluding as much transient detail from the processing as possible, to retain crispness. Adjust *Feel* to get the desired balance between tuning precision and realistic results.
- Adjusting *Input Ref. Pitch* can help minimize pitch jumping.
- If high-frequency pitches, such as the harmonics of string sounds, unfiltered synths or distorted guitars, are not being pitch-shifted, try increasing *PURIFY* and/or setting *Algorithm Mode* to it's different values.
- You can also leave the *Pitch Mapping Sliders* in place, and use *Xclude* mode to enforce a particular key/harmony - simply *Xclude* all pitches except for the ones you want the music to be playing on. Try the different *Xclude Round. Mode*s to see which method sounds best for your signal.

Tutorial F: Changing the Key/ Scale/Melody of a Recording using MIDI

Here's a run-down on how to use PITCHMAP with MIDI controlling the target pitch grid. How to set this up depends on which host you're using, please refer to the relevant section of this manual.

- Insert PITCHMAP and hook up MIDI control.
- Load the preset *01 MIDI Map Template - Medium* from the preset folder *02 MIDI Templates*
- Start the playback, and play some MIDI notes. The results are influenced by *Edit Repeat/ Visible/Custom*, the state of the sliders and the *Xclude Round. Mode* (see the relevant section of this manual)

The Standalone App

On MacOS X, PITCHMAP comes with a simple stand-alone application, the *Zynaptiq Stand-Alone*. This application is intended for quickly previewing audio through the plug-in, and is explicitly NOT intended to be an editing application. See it as a free bonus FYC.

The application allows playing files through PITCHMAP, as well as recording the results to a new file. To enable MIDI input, please activate *EXT MIDI*. As the application is not a full-blown host, it uses the Mac system audio preferences - including device & sample rate. As any open application including browsers, the finder etc can request to change the system sample rate, this means that in order to make sure your new file has the correct sample rate written, you should follow the following steps:

1) make sure you set the system sample rate to the rate that your source & destination files have/should have, from within the Audio MIDI Setup utility

2) start playback on the source file *before* you start recording

FAQ

Questions & Answers

Q: I am hearing ugly drop-outs/ring-modulation/distortion, what is causing that & what can I do?

A: Most likely, you are overloading your CPU. PITCHMAP is a computationally very "heavy-weight" process. The first thing you should try is to set the I/O and/or processing buffers of your host to higher values, ideally 2048 samples or higher. This will significantly decrease the CPU use and in most cases eliminates the artifacts mentioned above.

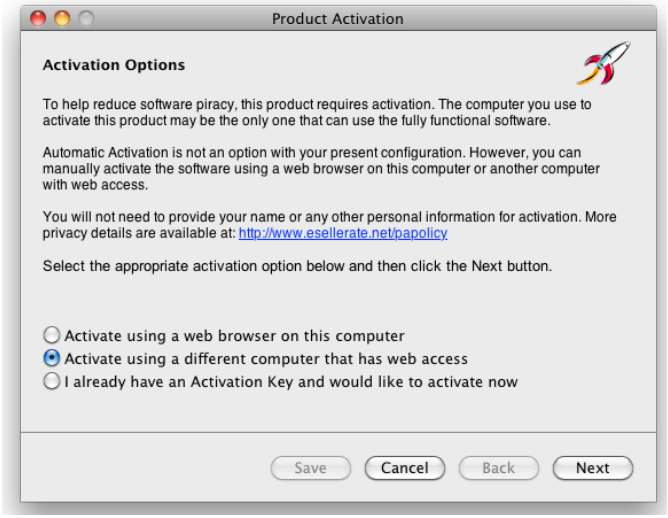
Q: (Mac) Logic Pro shows me a "Sample Rate Mismatch" or similar error when inserting PITCHMAP, and sound is distorted or stops altogether. What can i do?

A: You are overloading the CPU, but Logic is displaying the wrong error message. Increasing the I/O and process buffer sizes usually cures this issue. If not, see the CPU load optimization section below for some tips on how to get more out of your machine.

Authorization Issues Windows

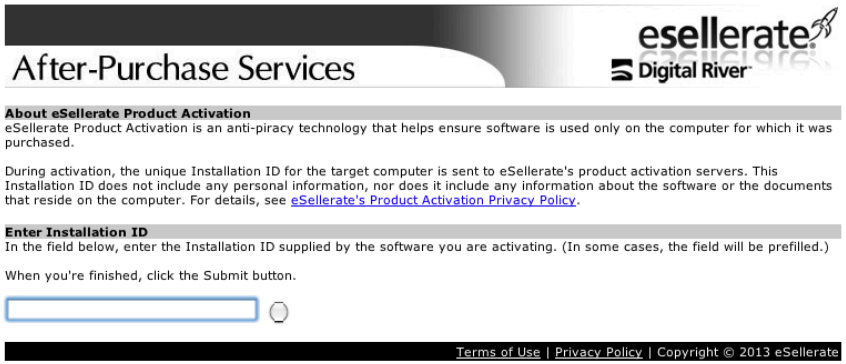
Q: I am using Windows and when I try to activate PITCHMAP, I am seeing an error message stating there has been an "Unknown Error" and activation fails. What can I do?

A: This means that there is something interfering with the activation process. Disable any firewall, security or anti-virus software and try again. If the problem persists you should temporarily disable your internet connection (by pulling the ethernet cable, for example). The Authorizer app will now display a dialog window that gives you several options for off-line activation:

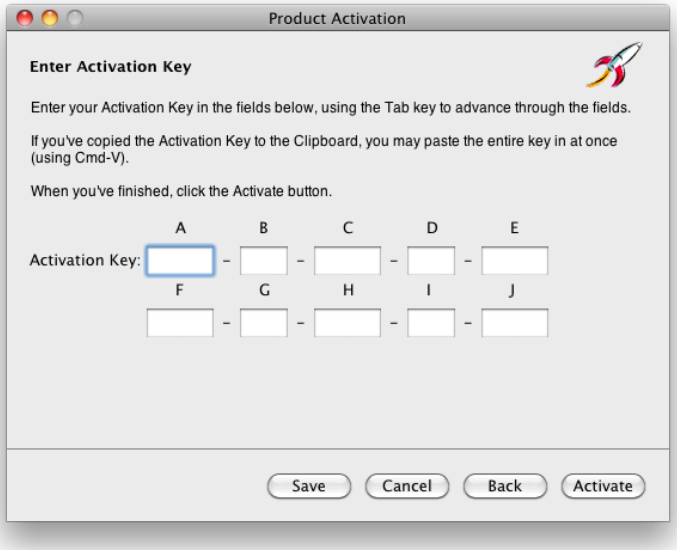


Choose "Activate using a different computer that has web access" and click "next". This will bring up another window.

Enter the Installation ID into the webform found at the URL displayed, using a different computer to access the web:



You will then receive a response code.



Enter the response code into the Authorizer app, click “activate”...and your software should now be activated successfully.

Sound Optimization

Here are a couple of tips on how to fine-tune the PITCHMAP parameters for best results.

- Increasing the *Threshold* parameter can help preserving drum transients
- Also, decreasing *Purify* below the default 50% can improve transient preservation
- If you're seeing a lot of "pitch jumping", try adjusting the *Input Ref. Tuning* and the *Xclude Round. Mode*
- *Bypass*ing very low and very high frequencies can help preserve bottom end impact and transient crispness; adjust the *Low-Cut* and *High-Cut* sliders so that any components you do not wish to tune/map are *Bypass*ed. Please note: when *MUTE* is on, this will completely remove those components instead of *Bypass*ing them!
- When working with signals that contain vocals, or if harmonics of certain sounds are being missed, try activating *Natural*
- If you are working on signals that are not very dense, such as single polyphonic instruments or vocal stems, you may hear some unexpected, higher-pitched "ghost copies" of the signal. Increasing *Electrify* slightly to around 60-70% will often reduce or completely remove this.
- In general, sound quality is proportional to pitch-shift factor, so you should always try to achieve desired harmony/melody changes with as little pitch-shift as possible. When using *MIDI MAP* mode, the easiest way to make sure you get the minimum shift possible is to *Reset* the GUI sliders before activating *MIDI MAP* and to use the *Repeat* mode. When using the *Pitch Mapping Sliders*, you can set their heads to *Round* by shift-clicking them, which sets them to a mode that automatically uses the shortest shift distance. Also, once you've set up a target key/scale using the sliders or the *Transform* macro, use the *Voicing Arrows* to define which source pitches get mapped to which interval in the target scale --- some voicings will use vastly less pitch shift and may sound just as good (or better)

- When working with very dense material, or material with very noisy or broadband sounds like heavily distorted guitar chords or aggressive synth-sounds, PITCHMAP may sometimes fail to recognize all harmonics and may move some of them to a residual layer (which typically holds drum transients and the like). In this case, you may hear some traces of the original pitch or inharmonic coloration. Raising *Purify* can reduce this effect.
- Raising *Purify* significantly will however make your sound resonant/synthetic. This can be counter-balanced to some degree by raising *Feel* or by adding small amounts of *Glide*.
- On some sources, like heavily detuned finger-picked acoustic guitars, pitch correction may sometimes not fully remove tuning inaccuracies even with *Threshold* and *Feel* at their minimum values. This is mainly due to a) beat frequencies in the chord and their harmonics being detected as separate pitches and being corrected differently than the rest of the sound and b) the very short pitch "bend" that the picking itself introduces by manipulating string tension. Try using *Strict* mode, which will remove more tuning inaccuracies while sacrificing some

CPU Load optimization

PITCHMAP uses a lot of CPU. The mathematics involved are highly complex, so there's a lot of computation to be done. However, CPU load is affected by a couple of variables. We'll outline strategies to get the most out of PITCHMAP on your CPU here.

1) **Use large buffer sizes**. PITCHMAP needs to look at a certain amount of signal to be able to discern which parts of the signal are the transfer function and which aren't. If your buffers are set to values that are shorter than that, CPU load increases significantly. PITCHMAP internally operates on a buffer size of 1024 samples, so in most host applications, you'll want to set the buffer size to the highest possible value. We recommend *at least* 512 samples.

2) In Logic Pro, all processes that are fed by a live source are processed on on CPU core. When using PITCHMAP under MIDI control, this means that all down-stream plug-ins are run on the same core as PITCHMAP whenever its software instrument track is selected and the record enable switch is lit. Selecting a different track in the arrange, or un-setting the record enable button allows CPU load to be spread out across all available cores, so this may help reduce CPU load.

3) Closing the plug-in GUI saves some CPU cycles, too.

4) Setting *Electrify* very low causes a higher CPU load due to more sounds being recognized, which causes the *Display* to render more symbols. Keep the value of *Electrify* around 50% if you don't explicitly use it for fine-tuning your sound.

5) Similarly, very high values for *PURIFY* increase CPU load - try backing it off a little.

Getting Support

If you’re having trouble with PITCHMAP that isn’t covered in these FAQ, please email us & we’ll get you up and running ASAP. We can be reached via http://www.zynaptiq.com/support-form/