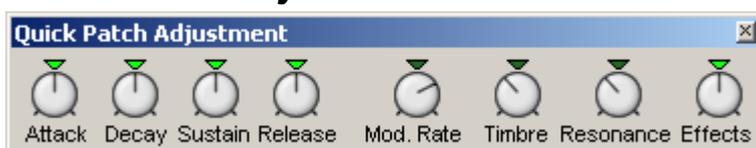


## The Patch Mutator and the Patch Adjustor

Since the OS version 1.4, the Nord Modular G2 Editor provides two new tools for patch modification, the Patch Adjustor for quick patch changes on a global level, and the Patch Mutator for extensive sonic exploration based on interactive evolution.

Together these tools greatly simplify the sound design process, and allow you to get interesting results even without knowledge of the inner workings of a specific patch.

### The Patch Adjustor



The Patch Adjustor contains a number of knobs for quick adjustment of all parameters of a certain category, such as all attack times or all resonances. In this way, you can quickly shape a sound in a more general way, without knowing the specifics of the patch.

For example, turning the *Attack* knob to the left will decrease all attack times in the patch, relative to their current position. Turning the *Timbre* knob will adjust filter frequencies, FM amounts, shapers and some other parameters that affect the timbre of the sound. If the result is not what you expected, just turn the knob back to its middle position, or click the centre marker. As soon as you move the focus to another variation or add or remove a module, the changes will be permanently applied, and the knobs will return to their middle positions.

When browsing a large number of patches, or when evolving new patch variations with the Patch Mutator (see the following section), this kind of quick changes can be very useful. A patch may be promising, but not exactly what you want. Then you can apply that last touch with the Patch Adjustor knobs. You do not need to track down where the specific parameters are situated in a complex patch.

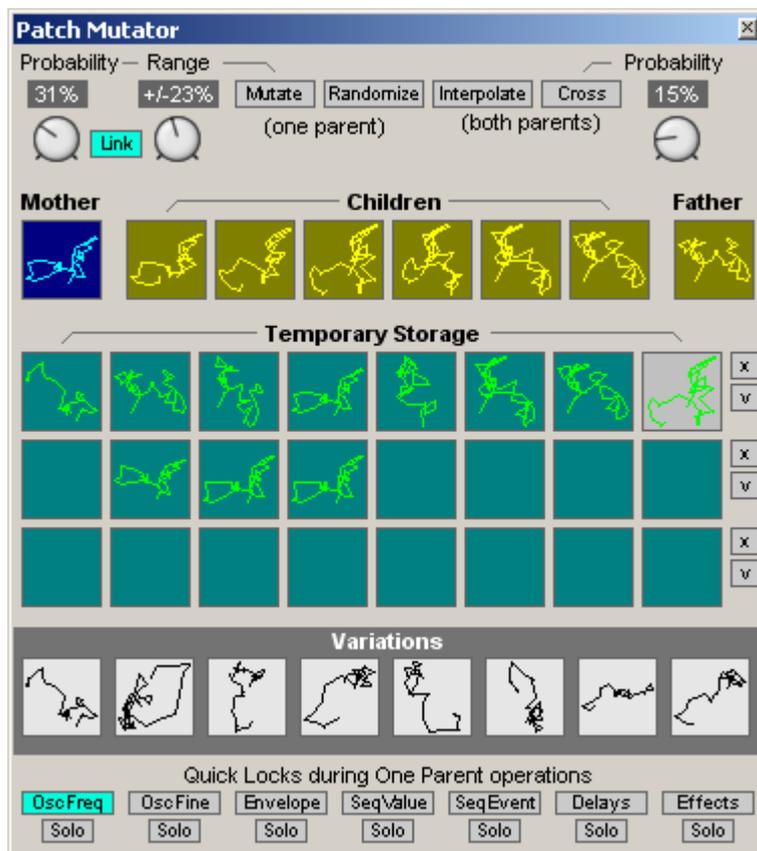
Patch Adjustor knob	Description
Attack	All envelope Attack parameters, as well as compressor, noise gate and envelope follower Attack parameters.
Decay	All envelope Decay parameters, as well as some Time parameters in the multi-stage envelopes.
Sustain	All envelope Sustain parameters, and some Level parameters of the multi-stage envelopes.
Release	All envelope Release parameters, as well as compressor, noise gate and envelope follower Release parameters.
Mod. Rate	All LFO and Random rates, and the rates of

	all effect modules incorporating an internal LFO, such as Chorus, Phaser and Flanger.
Timbre	Filter cutoff frequencies, FM modulation amounts, various waveshape parameters in oscillators and shapers, as well as various FX parameters affecting the timbre of the sound.
Resonance	All filter resonances, and Phaser and Flanger feedback.
Effects	Dry/wet effect parameters.

The Patch Adjustor lives as a floater on top of the editor window, like the Patch Browser, the Parameter Pages and the other tools of the G2 Editor. The Patch Adjustor is only available in the G2 Editor, not in the hardware synth itself.

It is important to understand how the Patch Adjustor knobs work. They are not to be considered as additional parameters in the patch or as additional Morph categories. They rather act as remote, relative editing tools for all parameters of a specific category, wherever they are located in the patch. Because of their different nature, these knobs cannot be assigned to MIDI Controllers or to physical knobs on the synth.

## The Patch Mutator



### Introduction

The Patch Mutator is a toolbox for creating new patch variations guided by your ear. It will help you explore different knob settings within a certain patch by simply listening and selecting the sounds you like. Most patches can generate a vast range of different sounds, but it is tedious and difficult to explore them manually, because of the sheer number of parameters in a patch, and because of the difficulty to predict the sonic results. Also, with the extreme flexibility of the Nord Modular G2, it is tempting to always look for new patches, and neglect the fact that one single patch often equates a full hardware synthesizer, with vast possibilities waiting to be explored.

In an interactive process inspired by natural evolution, the Patch Mutator creates a number of variations for you, based on one Mother sound, or based on a Mother and a Father, or more or less random. You listen to these new Children sounds one at a time, and select which one will be the parent of the next generation of children. This process can be repeated any number of generations, until you find the sounds you want. Usually you find lots of interesting sounds in the process.

The Patch Mutator is designed for speed and ease of use, with great care put into the interface design. In a few minutes you can create whole sets of new sounds, or series of interesting variations on existing sounds. The evolutionary approach makes it very different from simple patch randomizers available in some synthesizer editors. You

always work with a “family” of related sounds, comparing and selecting the best from each generation.

At any moment, a Child sound can be saved for future use in a Temporary Storage. Also, any sound from the Temporary Storage can be brought back into the breeding process as a parent. The G2 architecture allows eight variations to be stored with the patch, and any sounds from the temporary storage can easily be transferred to the eight variation buffers, and be saved as a patch in flash memory or on disc.

The main purpose if the Patch Mutator is encourage and simplify creative sound design, but there are many variations on this theme. A few suggestions:

- Quickly modify a sound in a studio situation where you want a specific sound.
- Explore the potential of your patches beyond the default values and your first tweaks.
- Explore other people’s patches without knowing exactly how they are constructed.
- As a composition tool: Create a number of related but different sounds or textures that can later be put together.
- Evolve complex sequences that control other equipment through MIDI.
- Evolve mappings from MIDI controllers to sound engines, to find new and interesting ways of performing.
- Evolve new sounds, loops and textures live, as a way of improvisation.

There are probably as many ways to use it as there are users. Consider it a powerful toolbox at your disposal for sonic exploration guided by your ears.

The Patch Mutator has been developed in collaboration with composer and researcher Palle Dahlstedt, based on his research in computer-aided creativity. Read more about the historical and theoretical background in the end of this chapter.

**NOTE! The Patch Mutator will turn the knobs and push the buttons for you, but it will not affect the connections between modules, and it will not add or delete modules.**

**NOTE! The Patch Mutator only exists in the editor, and is not available in the synth itself. Therefore, it can only be used when the G2 is connected to the computer editor.**

### **The Patch Mutator Window**

The Patch Mutator is a floater in the editor, just like the Virtual Keyboard and the Parameter Pages. It can be opened or closed by selecting Patch Mutator on the Tools menu (Shortcut: Ctrl-2). With the Patch Mutator floater open, you can still edit the patch as usual, add or delete modules or turn the knobs.

The Patch Mutator float consists of five different regions. From top to bottom:

- Various settings and action buttons for the different breeding operations that generate new children: Mutate, Randomize, Interpolate and Cross. See the Patch Mutator Reference section below for a detailed explanation.
- A row of boxes representing the parent and children sounds. Each box contains a “chromosome” graph. The shape of this curvy line is derived from the actual parameter values. The chromosomes show at a glance how different the variations are. In the screenshot above, the children are clearly quite similar, while the sounds in the temporary storage look completely different from each other.
- The Temporary Storage, which serves as a scratch pad during the evolutionary process. You can save sounds you like by dragging them to a box in the storage, and later you can drag them from the storage to one of the parent boxes in the row above, to continue breeding. A row from the Temporary Storage can quickly be transferred to the eight patch variation buffers by clicking on the “v” button to the right of the row. Click the “x” button to clear the row. Right-click and select Delete to clear a box.
- A row of eight boxes representing the eight variation buffers of the G2. Clicking on one of these boxes is the same as pushing a variation button on the G2. The variations can be dragged to the parent boxes or copied to the Temporary Storage.
- A row of Quick Lock buttons that are used to select which parameter categories are temporarily excluded from the random changes caused by the Mutate and Randomize operations. In many cases you want to vary certain parameters, while leaving others unchanged. For example, you may not want to affect the pitch of the oscillators in the patch. Then you uncheck this button, and they will be left unchanged by these operations. By clicking the *solo* button below a category, only one category is affected by the random changes.

## Using the Patch Mutator

The Patch Mutator can be used with all patches, also with any patch from previous OS versions. It works well with most patches straight out-of-the-box, while others may need some slight modifications and tweaking. Experiment with the *Quick Lock* buttons, and you will probably find a way to evolve interesting sounds with most patches. After a while, you will understand how to create patches that are tailor-made for evolutionary exploration.

You may want to exclude certain modules permanently from random changes, for example a part of a patch that has been carefully worked out and should stay as it is.

- To exclude one or more modules from the Patch Mutator operations, select the module(s), right click and check *Exclude From Mutations*.

When the Patch Mutator is visible, excluded modules are indicated by a thin red frame. The *Exclude From Mutations* settings are saved with the patch.

To simplify the use of the Patch Mutator, some modules are excluded by default. Most of these are modules that you probably wouldn't want to change in a random way, such as output modules, compressors, MIDI out modules or pitch trackers. If you want, you can include them by simply right-clicking and unchecking the *Exclude From Mutations* option.

## Quick Start

The Patch Mutator contains a range of very powerful tools for creating Patch variations, and it can be used in many different ways. Below is a short description of the tools available:

Tool	Description
Mutate	Mutations are small random changes in a sound, similar to the slight reading errors occurring in DNA replication in nature. The changes may be good or bad. You decide. Use this tool to create new variations based on one existing sound.
Randomize	Use this tool to create completely random patch variations. All parameters will get a new random value. Use this tool to find interesting starting points for further exploration.
Interpolate	Use this tool to create a set of variations that gradually progress from one existing sound to another.
Cross	In nature, DNA from two parents is combined in a partially random process called crossover. Basically different segments are copied from each parent and combined. Use this tool to create new variations based on random combinations of two existing sounds.

Try this step-by-step example of how to use the Patch Mutator to get a feel for what it can do. See the Patch Mutator reference section later in this chapter for a detailed explanation of each operation and further hints on how it can be used.

1. Load or create a patch you want to explore. It may be wise to start with a rather simple patch, maybe a sequencer patch, where you here the changes as they happen.
2. If the sound came with some Patch Variations, listen to them and select one that you think is a nice departure point.
3. Drag the selected variation to the Mother box and click Mutate to generate six new children (shortcut: double-click on the variation, or press E on the computer keyboard).
4. Listen to the children one by one, by clicking once on each box or moving the focus with the arrow keys. They will be slight variations of the parent sound. If you like one and want to save it, drag it to the Temporary Storage (shortcut: S).
5. Select the best of the children, drag it to the Mother box and click Mutate (or simply press E). Six new children are created. These are grandchildren to the sound you started with. If none of the children suits your taste, click Mutate again. Six new Children are created from the same Mother as before.
6. If a sound is good, but not quite there yet, you can try the Patch Adjustor knobs to make swift changes, for example shorten the attack times or adjusting the timbre. Then go on from the modified sound.
7. If the children differ too much or too little from the parent, try changing the Mutation Probability (few or many changes) and Mutation Range (small or large changes).

8. Repeat steps 3 to 5 any number of times, and save the sounds you like in the Temporary Storage.
9. Also try mating one sound with another. Drag one nice variation to the Mother box, and another to the Father box. Then press *Cross*. Six new children will be created that are random mixes of the two parent sounds. This works best if the parents are not too similar.
10. When you are finished, copy a row from the Temporary Storage to the eight variation buffers by clicking the “v” button to the right of the row, and save the patch.

Instead of starting with an existing sound, you can click Randomize to generate six completely random children. Some of these may be very ugly, or even totally quiet. That is part of the game. If none is good, try clicking Randomize again. For most patches you will soon find a sound that will work as a good starting point. Remember that the sound doesn't need to be perfect. Through the process of repeated breeding you can modify it to suit your taste.

You can manually modify any sound (child, parent or in the storage) at any time, either by turning knobs in the usual way, or by using the Patch Adjustor. If you breed from this changed sound, these changes will be inherited by the children.

**NOTE! All operations in the Patch Mutator work with multi-step UNDO in the usual way.**

**NOTE! The Temporary Storage is not saved with the patch. When the patch is closed, the contents are lost and cannot be recreated. Only the eight Patch Variations in the bottom row are saved with the patch. Make sure that you save your favorite sounds as Patch Variations (see below).**

## **Patch Mutator reference**

In this section you will find a detailed description of the different operations in the Patch Mutator, and some hints about how to use them in sound design.

There is only one Patch Mutator window, but under the surface there are one instance of the Patch Mutator per open patch. This means that each open patch has its own Temporary Storage, probability settings and Quick Locks. However, these settings are not saved with the Patch.

Normally the G2 has eight edit buffers, the Patch Variations, and one of them is always active. With the Patch Mutator, this is no longer the case. If a parent, child or a sound from the Temporary Storage has focus, no Patch Variation is selected. Instead, the Patch Mutator uses a ninth, internal variation. This means that you can evolve and experiment without touching any of the eight Patch Variations.

### **Which parameters will be affected?**

The Patch Mutator turns a lot of knobs for you, and it is important to understand which parameters are affected, and how you can permanently or temporarily lock or exclude some parameters or modules from being changed.

*Exclude from mutation*

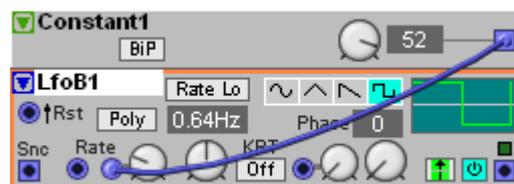
Many patches contain parts that should not be changed, either because you want to keep manual control over, say, your reverb settings, or because you have carefully designed a circuit that only will be messed up by random changes.

You can exclude any module or combination of modules permanently from being affected by the Patch Mutator. Excluded modules are marked with a red frame when the Patch Mutator is open. This property is saved with the Patch.

- Select one or more modules, right-click and check *Exclude From Mutation*.

The Exclude From Mutation setting can be used to optimize a patch for sound breeding, by locking all modules that will cause unwanted changes, and leave the rest for the Patch Mutator to explore. This is very useful when sharing patches with other people. They can start evolving immediately, without detailed knowledge about the patch.

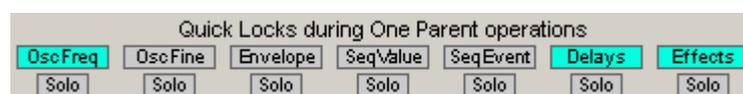
Sometimes you want to freeze only one or a few parameters of a specific module, but leave others for evolutionary changes. In most cases you can achieve this by locking the module and using an extra Constant module to modulate the parameter. By adjusting the modulation amount, you can also control the range of the mutations, as in this example:



*The LFO is excluded from mutations, but the rate will still be evolved within a specific range, controlled with the Rate modulation amount.*

### *The Quick Lock buttons*

You may not always want to evolve all kinds of parameters at the same time. Halfway through, maybe the sequencers behave very nicely, but you want to continue to evolve the other parts of the patch. Or you may want to temporarily exclude all effects from random changes without. Or the opposite – for a while you want to evolve only the sequencer values and sequencer events.



*The Quick Lock buttons. In this case oscillator frequencies, delays and other effects are locked.*

Using the Quick Lock buttons in the Patch Mutator, this kind of temporary parameter exclusions are easily achieved. There are seven categories, covering some of the basic parameter types. When a button is lit, parameters of that category are not affected by random mutations.

Below each button is a Solo button. Click this if you want to evolve only the parameters of a specific category. It is possible to activate more than one Solo button at the same time. Note that the solo buttons temporarily locks *all* other parameters,

also those that are not covered by the other Quick Lock buttons. If you click the SeqValue Solo button, only the levels from the various step sequencer modules in the patch will be evolved. Nothing else.

<b>Quick Lock Button</b>	<b>Description</b>
OscFreq	The main frequency parameter of various kinds of oscillators, often called Semi, Freq, Fact or Partial. Also DrumSynth MasterOsc and SlaveOsc frequencies. This category should often be locked when you evolve keyboard patches, to avoid unwanted transpositions.
OscFine	The oscillator fine tune parameter, often called Cent.
Envelope	All Envelope times and levels.
SeqValue	All value faders of the step sequencers, including the Note Sequencer and the Control Sequencer.
SeqEvent	All gate and trigger event buttons in the various sequencers.
Delays	Parameters of the various Delay modules.
Effects	Parameters of the modules in the FX group.

Please note that the Quick Locks only makes sense with the Mutate and Randomize operations. Interpolate will have to interpolate also locked parameters to provide a smooth transition between the Mother and the Father. On the other hand, the Cross operation takes all the parameter values for the Children from either of the parents, and hence does not perform any random changes of parameters. It only combines existing parameter values randomly from two sources.

### *Probability distributions*

All parameters are assigned a probability distribution to maximize the chance of good musical results. For example, an attack time of 45 seconds is seldom wanted. Hence, the attack parameter will tend towards lower parameter values. Likewise, extreme Oscillator Frequency settings are less probable, and Oscillator Cent parameters tend towards middle values.

### *Permanently locked parameters*

Also, there are a few parameters in different modules that are never possible to evolve. These choices have been made to optimize the Patch Mutator for use with any patch from your patch library, and to make the tool easier to use by excluding parameters that are unlikely to be evolved. For example, you wouldn't normally want to evolve the signal type (positive, bipolar, inverted positive etc) that can be set on the output of many modules, or the Mute or Bypass buttons on many oscillators, filters and effects. If you still want to evolve these parameters, you can mimic the behavior with other modules, and evolve the parameters on those. For example, bypass functionality can easily be created with a Switch 2-1 module.

### *Modules excluded by default*

A number of module types are excluded from mutations by default when you import old patches or create new modules. This is done to simplify the use of the Patch

Mutator for newcomers, and to make it easy to evolve new sounds with old patches without tedious modifications.

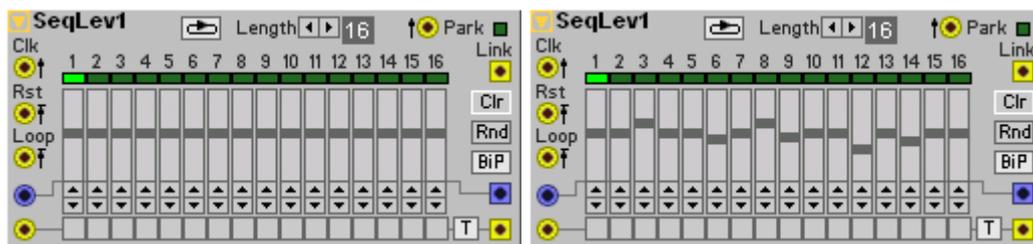
Examples of modules excluded by default are: Input and Output modules, most MIDI modules, the Vocoder, the 6ch Stereo Mixer and a few others.

If you want to evolve the parameters of any of these modules, simply right-click it and uncheck *Exclude from Mutations*.

## Mutate

The Mutate operation produces new Children from one parent by introducing random changes to the parameter values of the Mother, so called *mutations*, similar to the DNA reading errors in the biological reproduction process. The extent of the random changes can be adjusted from small, almost unnoticeable nuances to wild alterations approaching total randomization.

When you click the Mutate button, six new children will be created based on the Mother sound. You can adjust the mutation probability, and the maximum range of the random changes. These are linked by default, so that a small mutation probability will lead to few but large random parameter changes. A high probability will lead to many, but very small random changes. You can adjust them individually by unchecking the *Link* button between the knobs.



*Mother (left), and a Child produced with default Mutation Probability and Mutation Range.*



*Two Children produced by Mutation, with low Mutation Probability and high Mutation Range (left), and high Mutation Probability and low Mutation Range (right)*

Parameters that are excluded from random changes, either in excluded modules, or parameters locked with the Quick Lock buttons, will be copied from the parent sound without changes.

## Usage hints

The primary use of the Mutate operation is to create variations on existing sounds. Start with rather high probability and range settings to explore the possibilities. When

you find a sound you like, you can use smaller and/or fewer mutations to explore the nearest vicinity of the sound.

If you find it difficult to evolve all parameters at the same time, lock some categories with the Quick Locks and evolve a subset of the parameters.

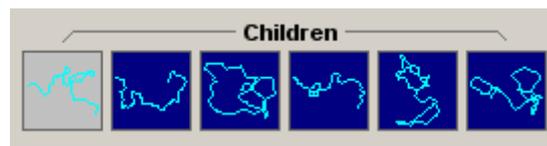
### *Shortcuts*

Mutate from any sound in the Patch Mutator by double-clicking on it, or select it and press the E key on the computer keyboard. The sound is copied to the Mother position and six new Children are produced.

To produce six new Children from the current Mother, press the U key on the computer keyboard.

### **Randomize**

Six new children will be created by generating completely random values for all parameters that are not excluded by the right-click menu option or by the Quick Lock buttons. The values of the excluded or locked parameters will not be randomized, but instead copied from the sound that has the focus when the Randomize operation is performed.



*Random Children bear no resemblance to each other.*

Randomize often produce wild and crazy sounds, but with careful use of the Quick Lock buttons and module exclude, you can control which parameters will be affected.

If you want only slight random variations on your sounds, use Mutation instead (see above).

### *Usage hints*

The Randomize operation is wild, and can create really strange sounds. Sometimes they are completely quiet, or very soft, usually because some mixer level was set very low by accident. But don't give up. For most patches you will soon find interesting and very unexpected sounds, that may serve well as starting point for further breeding with the Mutate and Cross operations.

Also try Crossing a random sound with a nice Patch Variation. In this way you can create new sounds that are not completely weird, but still new and fresh.

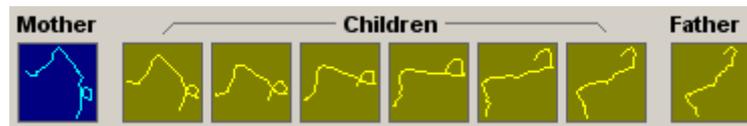
### *Shortcut*

Press N on the computer keyboard to create six new random Children. Locked and excluded parameters will be copied from the currently focused sound.

### **Interpolate**

Six new children will be created by interpolating all parameter values linearly between the Mother sound and the Father sound. The children will show a step-wise

progression from left to right. This is clearly visible in the chromosome graphs, which will morph between the two parents' graphs.



*The result of an Interpolation between two sounds. Note how the graphical representation of the sound gradually “morphs” between the Mother and the Father.*

Parameters with a range of only a few values (e.g., sequencer events or radio buttons) will switch at slightly different times, to create a more interesting result.

**NOTE! Interpolate can change the values of all parameters in the patch, regardless of the lock buttons and module excludes. However, parameters that have the same value in the father and the mother will not change.**

### *Usage hints*

This tool is useful to create smooth progressions between two different sounds. If you think the steps are too large between the Children, you can further interpolate between them.

### *Shortcuts*

While holding Shift on the computer keyboard, drag and drop one sound on another to perform an Interpolation between them.

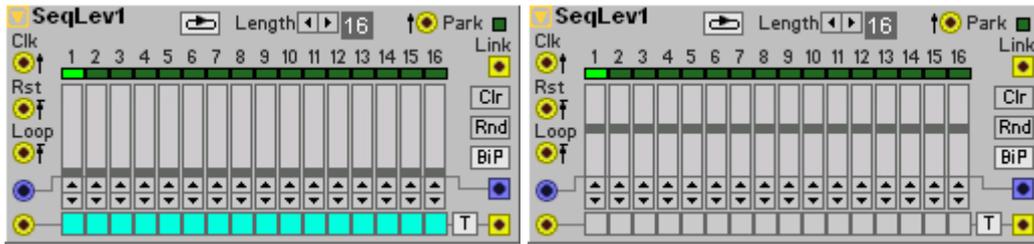
You can also perform an Interpolation without using the mouse. With the Patch Mutator active, select a variation you want to use as parent. Then copy it to a parent box with the O key (Mother) or the T key (Father) on the computer keyboard. When both parents are in place, press I to perform the Interpolation.

### **Cross**

Six children will be created by copying each parameter value from either the Mother or the Father sound. This is similar to the merging of DNA from two parents in natural reproduction. It works like this:

The parameter values are copied from one of the parents. At random points in the process the copying process starts reading from the other parent instead. In this way, the copying jumps back and forth between copying from the mother and copying from the father. The result is a sound where the parameters come from both parents, in a random combination.

With a higher *Cross Probability* setting, the copying will switch more often between the two parents during the copying process. In different words – a small cross probability will result in large sections of the patch being intact from one parent, for example a whole sequencer module, or a few neighbor modules. This will result in children more obviously similar to the parents. An example:



*Mother and Father.*



*Their Children, produced with low (left) and high (right) Cross Probability.*

During the crossover process, no random changes are made to the parameter values. All parameters of the children will have the value from either the mother or the father.

### *Usage hints*

Cross works very well when the parents are rather different. The Children will differ in the same way as brothers and sisters do – the nose from Father, the eyes from Mother. The rhythms, timbre or other qualities of the sound will be clearly identifiable as coming from either parent.

Since Cross doesn't introduce any random changes, it can be used with tricky patches that are not so well suited for mutations and randomization. This can be the case with FM synthesis and other synthesis techniques where the internal relationship between the parameters is crucial. Since it uses only "safe" values, there is a good chance that the children will be good.

Try mating two nice Patch Variations, to produce more nice and well-behaved variations. You can also try to mate a nice sound with a weird random sound to induce some new life into too well-behaved variations.

Mating siblings (two sounds from the same set of children) usually doesn't work well. Usually siblings are already genetically quite similar, so the resulting sounds will not exhibit much interesting variation.

### *Shortcuts*

While holding Ctrl on the computer keyboard, drag and drop one sound on another to perform a Crossover from those two sounds.

You can also perform a Crossover operation without using the mouse. With the Patch Mutator active, select a variation you want to use as parent. Then copy it to a parent box with the O key (Mother) or the T key (Father) on the computer keyboard. With the parents in place, press X to perform the crossover.

## The Temporary Storage

With the Patch Mutator, it is easy to produce a vast number of sounds, and often one doesn't know until later which sounds should be kept. Temporary results and unexpected sounds can be stored in the Temporary Storage, to be reused later in the breeding process.

To save any sound (parent, child or variation) in the Temporary Storage, drag it to any of the 24 storage boxes. If box already contains a sound, it will be overwritten. You can also move a sound within the storage by dragging it to a new box.

Click the "x" button to the right of a row of sounds in the Temporary Storage to clear the whole row. Right-click a box and select Delete to clear it.

If you accidentally overwrite or erase a sound, remember that Undo (Ctrl-Z) works with all operations in the Patch Editor.

## Saving Sounds

The Temporary Storage is not saved with the patch to flash memory or disk. Only the usual eight Patch Variations are saved with the patch. However, you can easily transfer your favorite sounds to the eight patch variations and save the patch.

To save a set of eight sounds from the Temporary Storage, arrange them in a row in the storage with the mouse. Then click the "v" button to the right of the row. The whole row will be copied to the eight patch variations (overwriting the current variations). Then you can save the patch in the usual way, to flash memory or to disk.

**NOTE! If none of the eight Patch Variations is selected during Save, the Patch will open with the first Patch Variation selected. This means that the Patch may not sound exactly as it did when you saved it.**

## Patch Mutator Shortcuts Overview

Keyboard Shortcuts	Action
Ctrl-2	Show or hide the Patch Mutator
Ctrl-3	Show or hide the Patch Adjustor
1	Set focus on Mother
2, 3, 4, 5, 6, 7	Set focus on one of the Children
8	Set focus on Father
Arrow keys	Move focus around in four directions
O	Copy focused variation to Mother
T	Copy focused variation to Father
E	Copy focused variation to Mother and create new Children by Mutation
U	Create new Children by Mutation from current Mother
N	Create new Children by Randomize
I	Create new Children by Interpolation between current Mother and Father
X	Create new Children by Crossover from current Mother and Father
S	Save focused variation to first empty box in the

	Temporary Storage
<b>Mouse Shortcuts</b>	
Click on a box	Set focus
Double-click on a box	Copy focused variation to Mother and create new Children by Mutation
Drag-drop a box to Temporary Storage	Save a variation in the Temporary Storage. Previous contents of that position are overwritten.
Drag-drop a box to a Patch Variation	Copy a sound to one of the eight Patch Variations. Previous contents of the variation are overwritten.
Shift-drag-drop	Copy the source and destination sounds to Mother and Father and create new Children by Interpolation.
Ctrl-drag-drop	Copy the source and destination sounds to Mother and Father and create new Children by Crossover.

## Historical background

The Patch Mutator is based on evolutionary algorithms, a subdiscipline of the research field of Artificial Life. In this field, natural processes are studied and simulated, based on the conviction that we can learn from nature, both about how it works and how to perform other tasks more efficiently inspired by these processes. This has led to a deeper understanding of the complex systems in nature, but also to new algorithms for many computational tasks, such as optimization and searching.

Evolutionary algorithms were first proposed by computer science pioneer John Holland in the 1970s, in a form he called the “genetic algorithm”. Inspired by Darwinian evolution, it applies random variation and selection as an optimization technique. However, when working with aesthetic objects like images and sounds, it is impossible to define any formalized selection criterion. Nobody can define what is a beautiful sound!

In the middle of the 1980s, the British biologist and writer Richard Dawkins proposed a variation on the genetic algorithm, where a human does the selection based on whatever his aesthetic preferences are, a kind of interactive evolution. Others further elaborated upon this technique, for example computer graphics wizard Karl Sims, who designed a system for interactively evolving stunning computer graphics images. Since then, it has been used in numerous graphical applications, but not nearly as much in music, maybe because it is much easier to browse and judge a bunch of images than a bunch of sounds or compositions.

The G2 Patch Mutator is based on the work of Palle Dahlstedt, Swedish composer and researcher. His research interest has primarily been in the application of evolutionary algorithms as creative tools, aiding the search of novel musical expressions. This has resulted in a series of tools for evolving electronic sounds and musical scores, and also in programs that compose music by themselves using evolutionary techniques.

## Further reading

There are many books and articles published about evolutionary algorithms in different contexts. Below are references to a few texts relevant for the Patch Mutator, and also some introducing the field in general.

*Richard Dawkins: The Blind Watchmaker. Longman Scientific and Technical, 1986.*  
The book that introduced the concept of interactive evolution.

*Karl Sims: Artificial Evolution for Computer Graphics, 1991*  
Seminal paper on artistic use of interactive evolution by one of the pioneers, available online at: <http://www.genarts.com/karl/papers/siggraph91.html>

*Stephen Levy: Artificial Life – A report from the frontier where computers meet biology, 1992*  
A popular science book introducing the broad field of Artificial Life.

*Palle Dahlstedt: Sounds Unheard of – Evolutionary algorithms as creative tools for the contemporary composer. Chalmers University of Technology, 2004*  
A doctoral dissertation about the techniques behind the Patch Mutator, and other musical applications of evolutionary algorithms. Contains a historical and theoretical introduction, a series of papers and sound examples. It is available online at: <http://www.id.gu.se/palle/dokt/soundsunheardof.htm>