Using Hyper-instruments for the re-distribution of the performance control interface

Reprinted from the proceedings of the International Computer Music Conference, Arhus, 1994

Tim Anderson, Drake Music Project. TimAnderson@DrakeMusicProject.org

Debbie Hearn, Drake Music Project

Abstract

'Hyper-instruments' are large scale musical systems which enable the control of complex musical events by a performer. They can thus allow various aspects of the performance interface and its relationship with the operation of the musical system to be altered. This can allow a re-distribution of the performance skills required which is especially relevant to disabled musicians - augmenting their motor skills when performing musical material within a pre-constructed framework. Hyper-instruments thus facilitate a moveable boundary between preparation and performance according to the ability and experience of each person.

1. The use of Hyper-instruments to re-distribute the performance interface

All musical instruments require some kind of physical interaction between the player and a performance interface into the instrument. This interaction is now practised with acoustic instruments as a highly developed conventional art - a complex relationship which transforms physical actions into musical sound. 'Hyper-instruments' consist of large scale musical systems which can be designed to respond to performance control in a complex manner. A complex sonic response may thus be generated by a musical performance which can involve unconventional combinations of gestures and modes of control. Such a system is acting as a new, more complex instrument.

This makes them highly suitable for use by disabled musicians who typically have different combinations of musically precise movements to a non-disabled performer. Such systems can facilitate the control of musical events which are more complex than could be managed by human performance actions, and can thus allow various aspects of the performance interface and its relationship with the operation of the musical system to be altered.

Such modification of the performance interface facilitates a *re-distribution* of the performance skills required. This redistribution is especially relevant to disabled musicians - augmenting their

motor skills to facilitate the performance and control of musical material within a pre-constructed musical framework (whether performing live or as part of a composition process).

The concepts involved in the data processing modules used in *constructing* such systems are well known: this paper describes a use of such flexible high-level performing tools to enable the performance interface to be customised. For example, the number of degrees of freedom required for a particular physical performance gesture can be deliberately reduced, but this reduction can be compensated for by an increase in the *precision* of the movement. This increased resolution can then be used to control pre-prepared material or processes within the instrument which can increase the density or complexity of musical output for a given control input, thereby replacing the musical interest which might be lost by a reduction in other aspects of performance control.

2. Construction of Hyper-instruments

2.1 Hyper-instrument structure

Hyper-Instruments consist of three notional components: (a) Some kind of physical interface for the input of control data from the performer into the instrument; (b) Processing of this performance data; (c) The generation of sound output by musical processes and structures, or sound processing - controlled by the performance data.

2.2 MIDI and modular systems

The MIDI communication standard has enabled the codifying of elements of a performer's actions in a standardised manner. MIDI is an appropriate communication medium for a single performer, with (just) adequate performance bandwidth, and the advantages of standardisation. Thus, proprietary subsystem components (inter-communicating via MIDI) can be mixed and matched as appropriate to construct a Hyper-instrument. Components will include: (a) software to store, generate and process data; (b) synthesis and sound-processing devices; (c) MIDI performance controllers; (d) computer data entry devices which are not often considered as performance tools. This paper thus also acts as a review of practice which is developing the expressive capabilities of MIDI in performance.

2.3 The processing and music generation subsystems

A Hyper-Instrument's musical processing control aspects may exist as a single (non-distributed) system: either custom-built hardware, or a custom-designed complex software application, whether created using higher-level environments (eg Max, Sybil, Interactor), or a more general purpose language with musical extensions (eg MidiLisp, HyperMidi). Alternatively, a Hyper-Instrument can be constructed by assembling a number of disparate *simpler* components which are readily available. These may be hardware processors (eg MEP4), or facilities available within proprietary software applications (eg MidiGrid, Cubase). Each has its advantages:

The latter allows relatively easy adaptation (in the field) to each performer's requirements and abilities, and enables *modification* as a performer's abilities (physical or 'musical') change, eg with

practice. This facilitation of adaptation is important if the aim is to provide Instruments which can be used in the world of music at large (rather than inside institutions with software and technical support) by disabled musicians. Such systems need to be maintainable, cost-effective, and - most importantly - be understandable and *alterable* by musicians themselves, or their assistants. In contrast, a typical unified software solution will usually be less amenable to performers do not have considerable programming ability - a relatively rare combination.

3. The components of a Hyper-instrument

3.1 Performance data interface

Input devices can include: (a) Computer input devices such as the tracker ball, mouse, joystick, drawing tablet, touch-screen, or the console keyboard itself. These can be used with the foot, or with a very small but precise movement of the hand. Infra-red head direction detectors to emulate mouse movements are also available; (b) Various types of switches, pressure pads of varying sizes (mm to m) and sensitivities (cm to μm); (c) Pressure transducers (eg piezo-electric "bugs"); (d) Proximity detectors or light intensity detectors; (e) MIDI performance controllers, typically using single notes or regions on a MIDI keyboard, or pads on a MIDI percussion controller. Again the foot is often used. (f) Ultrasonic distance sensors (eg ' Soundbeam' ' Midi Gesture') which produce MIDI events from movement within or into the beam. The range of discriminable movement can be m to mm.

3.2 Processing of input performance data

The input data (eg from various physical movements or gestures) can be processed in a variety of simple ways such as (a) filtering. For example, a *single* direction of movement may be ' parsed out' from a two-dimensional movement of a tracker-ball (ie movement on one of the axes will be ignored); (b) scaling, or other simple arithmetic operations; (c) merging of data from different performance inputs (from the same or different performers) into a single piece of musical control data. For example, one performer can trigger notes at the right time, while another controls the pitch (say one of 8) by position in a sound beam several metres long. A third performer could be moving a finger within a 2cm range over a proximity detector to control the velocity, timbre or volume. More complex (typically software-based) analysis of speed direction or acceleration can also be performed, as exemplified by the Radio Drum. For example, an event can be triggered by a *reversal* of movement.

3.3 Control of music and sound output

Various operations can be carried out to convert performer actions into musical output: (a) Single events (notes, chords, phrases etc) can be pre-composed and stored. These can then be selected and triggered by performance actions (as described above). However, there may be an element of predictability in being restricted to triggering fixed material; (b) Pre-stored material may be *mutated*. Eg, notes can be transposed, chords can be transposed or inverted, and phrases can be transposed, inverted or retrograded, or different sections triggered, or looped etc; (c) Additional

material can be added, for example creating additional notes with various fixed or variable relationships with the 'input' note. Such alteration may be pre-selected or controlled by performance inputs, whether from another performer or another aspect of the performers movements; (d) Timing can also be synchronised to other material being played (perhaps by others in the group, or other phrases etc being played at the time within the same Instrument). For example the onset times of triggered material can be quantised to fit with other events being played (eg by MidiGrid); (e) Audio processing of the sound can be performed. For example the selection of reverberation style, chorus, or other 'effects' can be controlled, or the *parameters* of such effects can be altered.

4. Conclusions - the music preparation / performance boundary

Hyper-instruments allow the boundary between *preparation* work (pre-composed phrases, chords and structures, and a plan of their performable interrelations etc) and *performance* work to be *moveable* according to the balance of performance and compositional ability and experience of a person. A more severely disabled performer can spend more effort in an 'off-line' preparation of material and systems to enable the performance to then proceed with *fewer* or less precise physical performance inputs.

However, the more complex the instrument, the more experience and awareness is required by the performer of the *effects* of each input action. In every case, musical choices are required, and care and skill in instigating them is necessary in order to create a musically coherent and satisfying result. For example, even if the onset times of triggered events is heavily quantised, the *choice* and rough time of events is still significant.

Thus, software and hardware elements of Hyper-instrument systems allow the musical effect of performance gestures to be more complex. However, the feedback of the musical output in influencing the performer's appreciation of the relationship of the musical output with his/her performance input is still vital - such systems can still produce boring results if they receive unconsidered performance input.

For example, as described above, several performers can interact to *jointly* produce a performance - each has fewer physical parameters to consider, but musical awareness and ability is still needed for a 'good' performance.

Such instruments may still be as difficult to learn to play *well* - the overall performance bandwidth of the interface is maintained - but its components are *re-allocated* to different aspects of performance control. Thus, complex music can be performed by musicians with a wide variety of types of abilities.

NB. Examples and diagrams were omitted from this paper by force of space restrictions, but are now appended below.

Using Hyper-instruments to re-distribute the performance control interface

Tim Anderson
Dept of Electronics, University of York, York, UK

Debbie Hearn
Drake Music Project Research Group, Ripon, N. Yorks, UK

'Hyper-instruments' are large scale musical systems which allow an arbitrary relationship to be specified between performance control and musical output. They are normally used to facilitatate the control of complex musical structures by a performer.

However, as the relationship between the performance interface and the operation of the musical system can be altered, this can facilitate a *re-distribution* of the performance skills required to control the instrument.

This is especially relevant to musicians with physical disabilities - a Hyper-instrument system' s control interface can be mapped more closelyand effectively to each performer' s combination of motor skills.
In addition, a person' s ability to perform musical material in real-time can be augmented by the use of pre-composed musical material or processes within a pre-designed framework.

Hyper-instruments thus facilitate a moveable boundary between preparation and performance according to the abilities and experience of each person, both physical and musical.
They can also allow the task of controlling an instrument to be *divided*, allowing two or more people to work together; each controlling a single aspect of the instrument.
Thus music can still be created and performed by people who have limited range or precision of movement

Key aspects of Hyper-instruments used to re-distribute the performance control interface

Flexible physical control interface: *detecting* a performer' s movements

Using a variety of physical transducers to input a performer' s movements as control data:-

Eg Wheels, mouse, joystick, computer console, drawing tablet, touch screen, pedals, sonic beams, pitch detectors, switches, light beams, pressure pads, breath controller etc.

Flexible movement processing: *interpreting* a performer' s movements

Using a variety of processing algorithms to convert the performer' s input data (movement) into performance control signals:-

Eg Processing of position, distance, strength, velocity, acceleration, direction, or style of movement (singly or in combination).

Flexible music processing: *converting* performance control into musical output

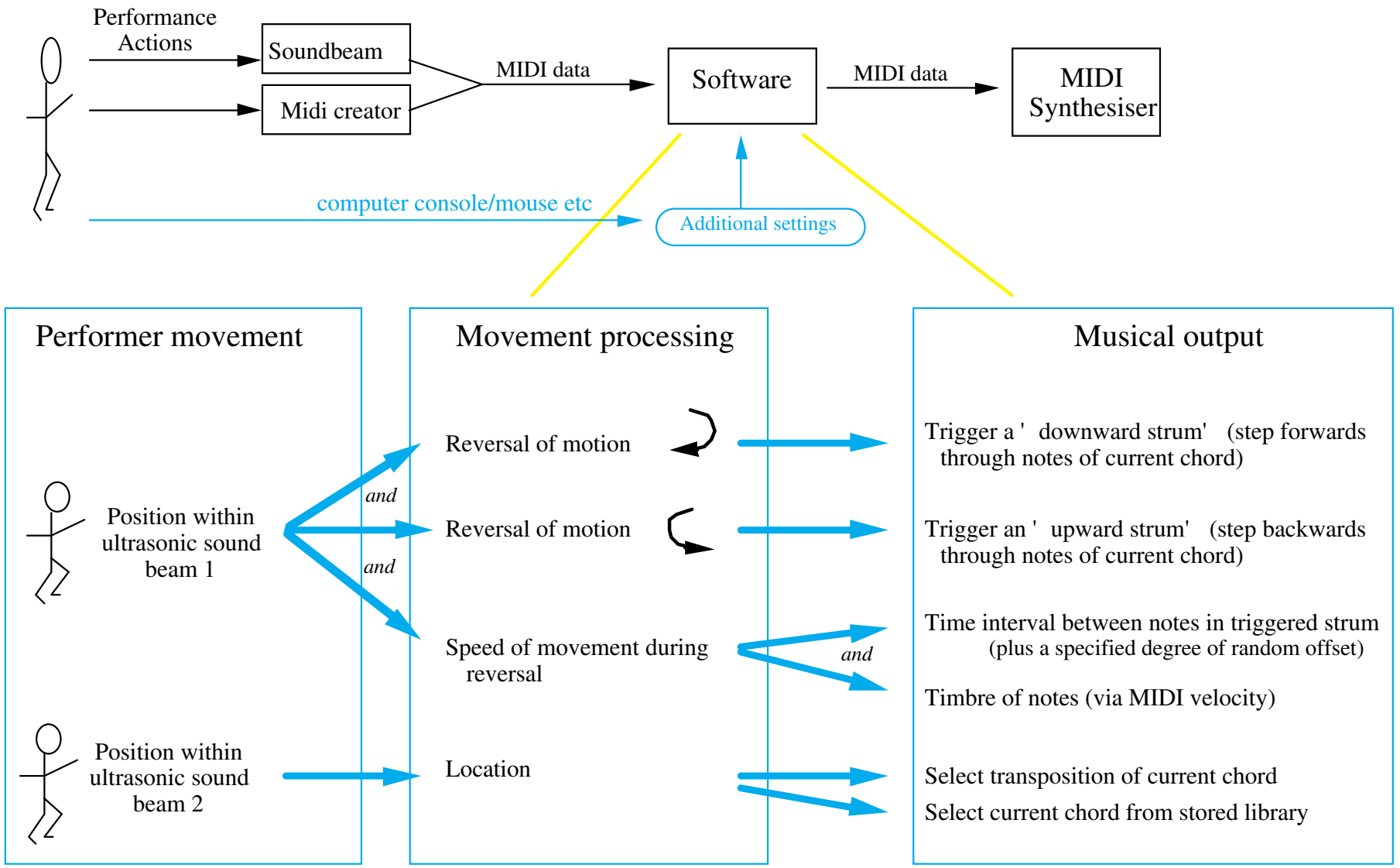
Using a variety of operations to convert one or more performers' control signals into musical output, by controlling the creation, mutation or processing of musical or sonic material:-

Material can be atomically specified or created (eg notes, chords, note rows, phrases etc) with a variety of relationships (eg time or pitch) with the performance control signals.

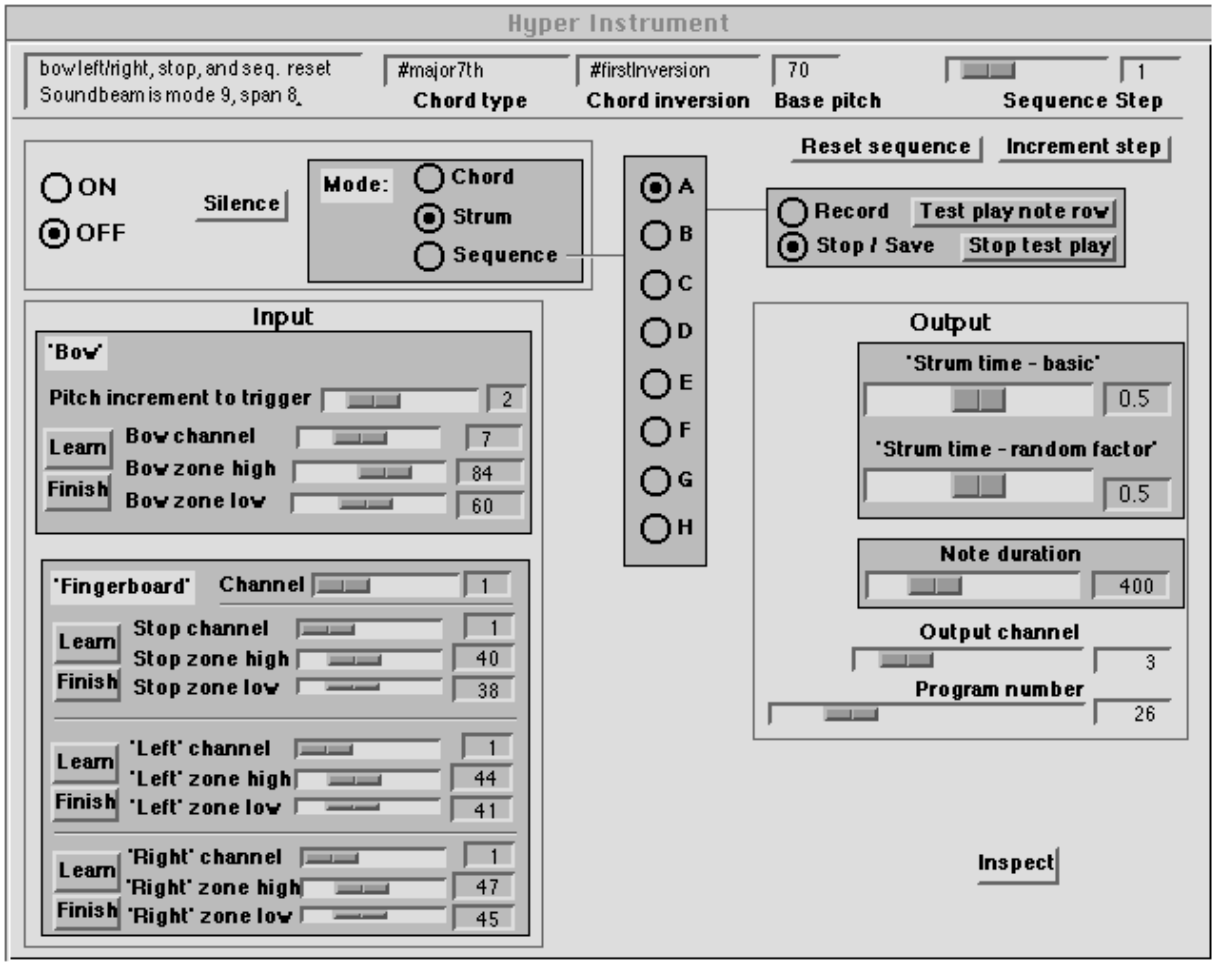
variable relationships), loop points selected, time relationships altered, or synchronised to other material.
Audio processing of the sound can be selected or altered.

Construction from available, cheap and maintainable components

Hyper-instrument Example 1 Using two ultrasonic beams to strum a guitar

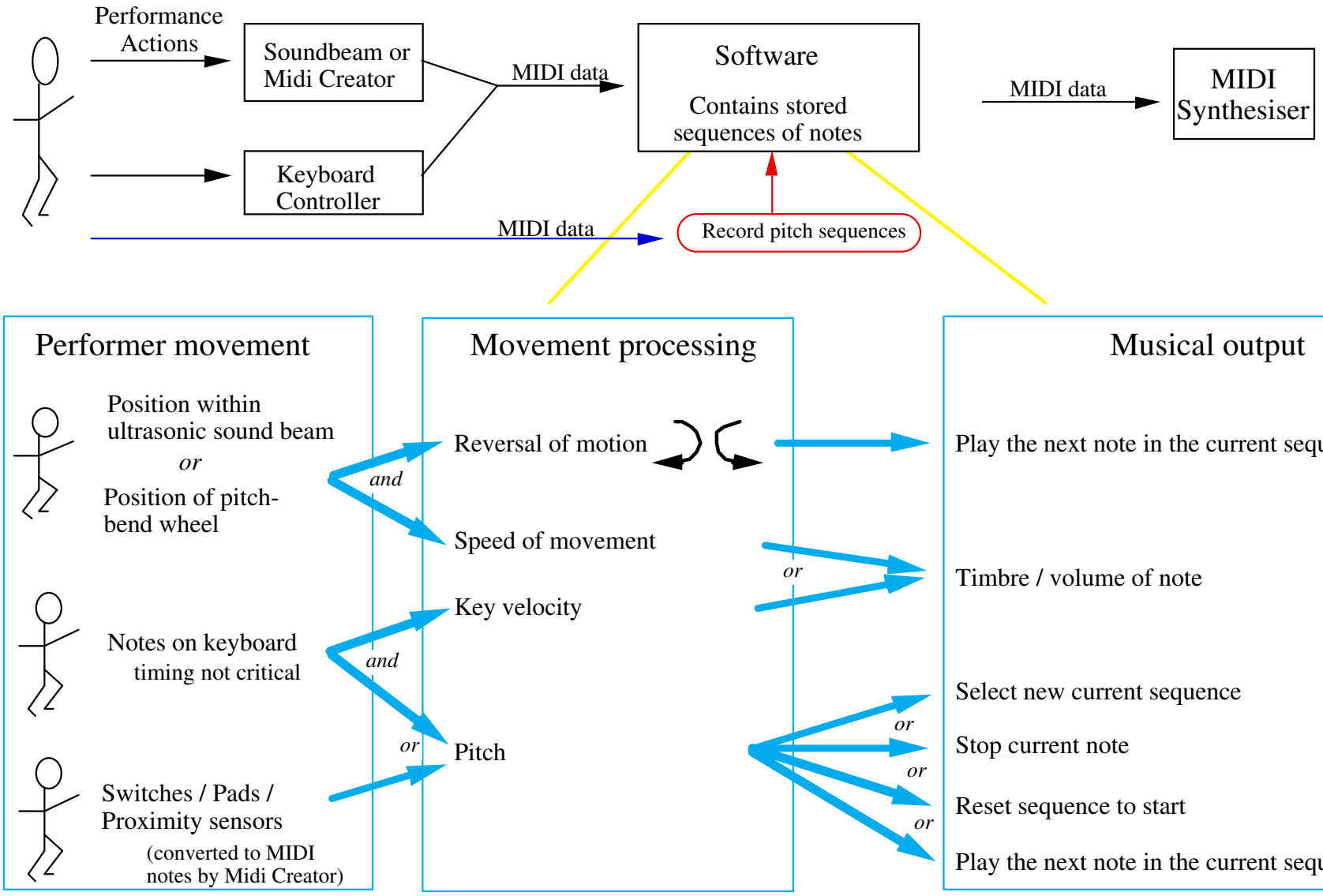


Hyper-instrument Example 1 - Software Control panel



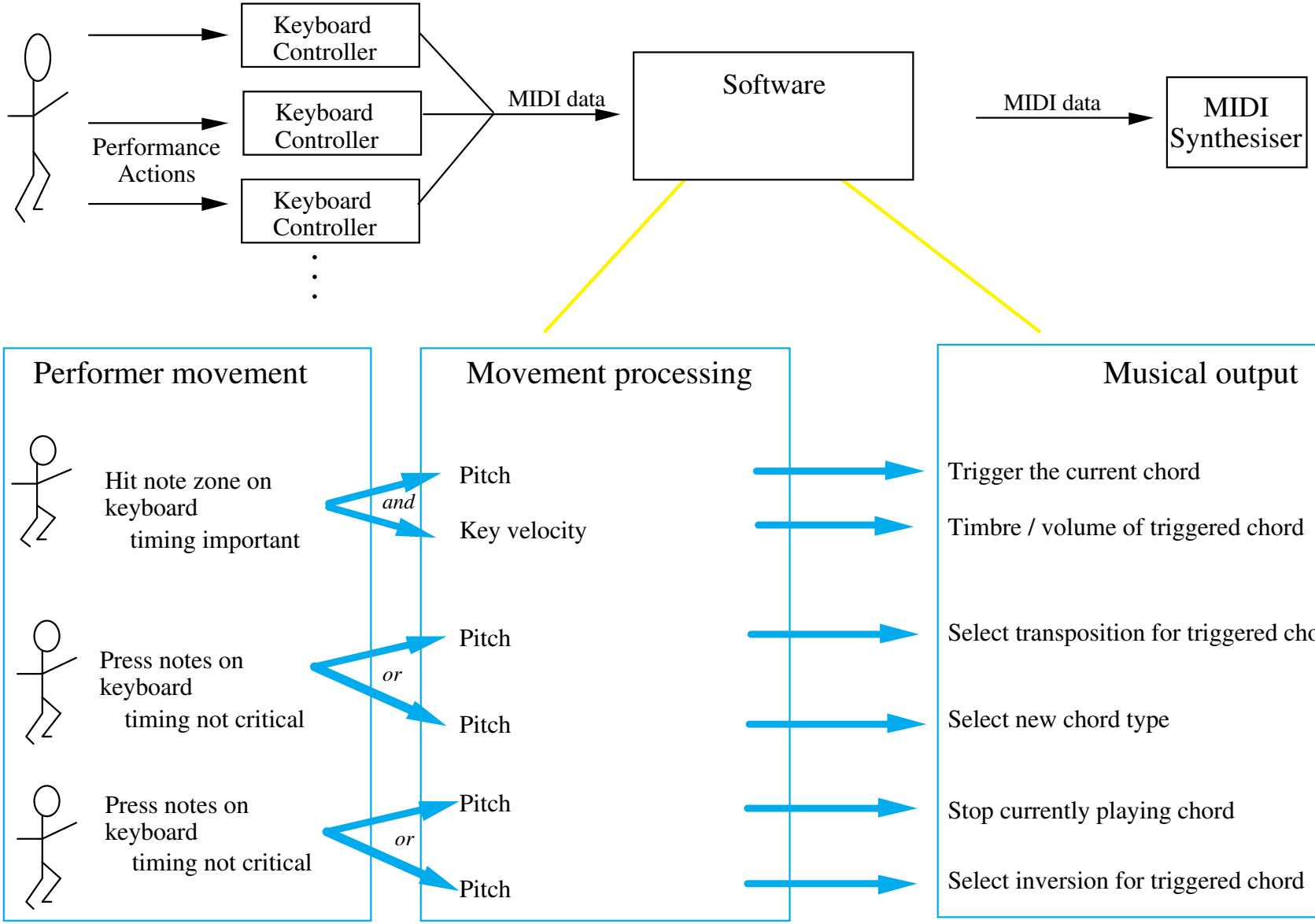
Hyper-instrument Example 2

Using single notes, wheel, switches or ultrasonic beam to play a set of violin cadenzas



Hyper-instrument Example 3

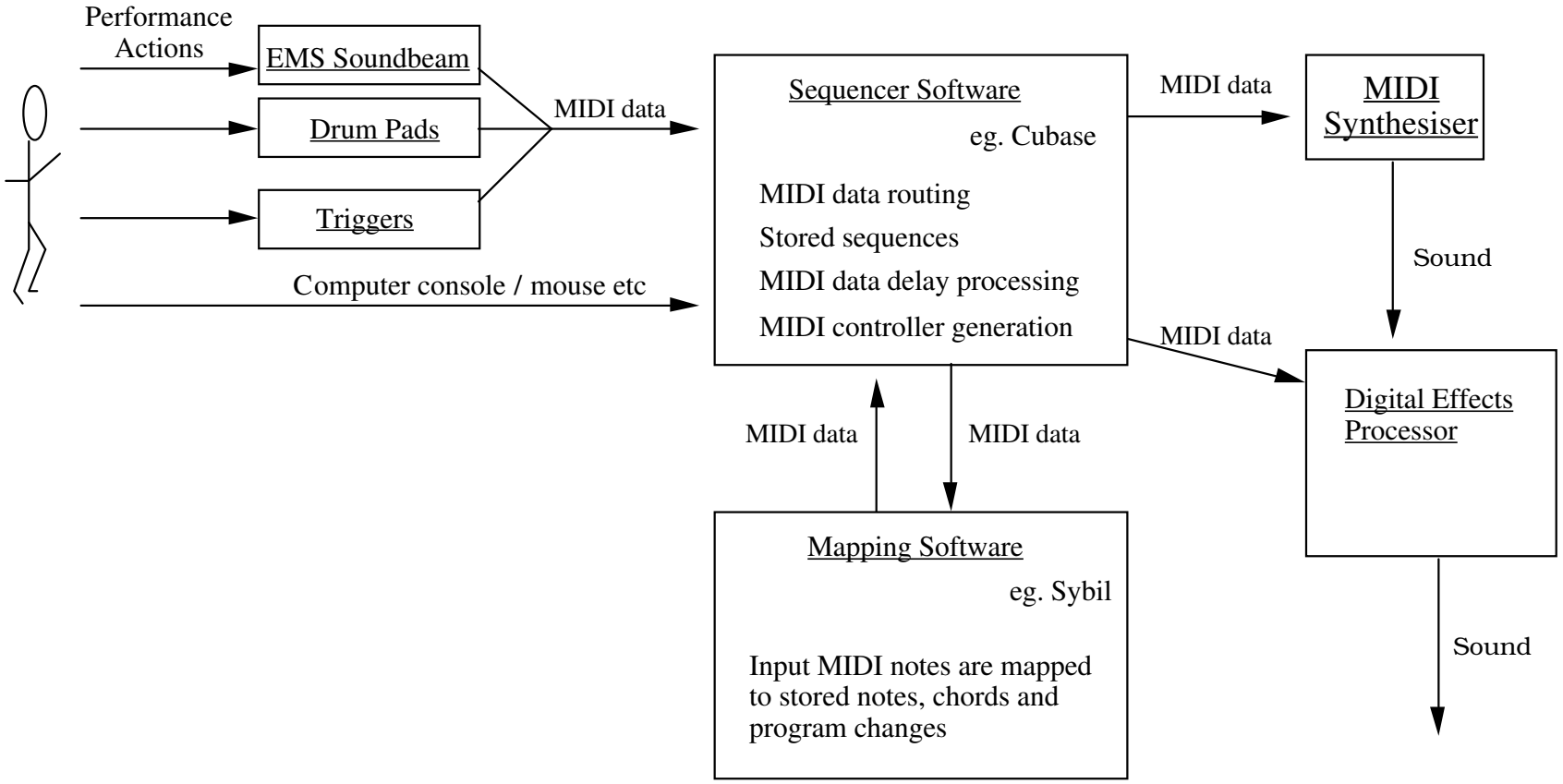
Using notes on several keyboards to play chords

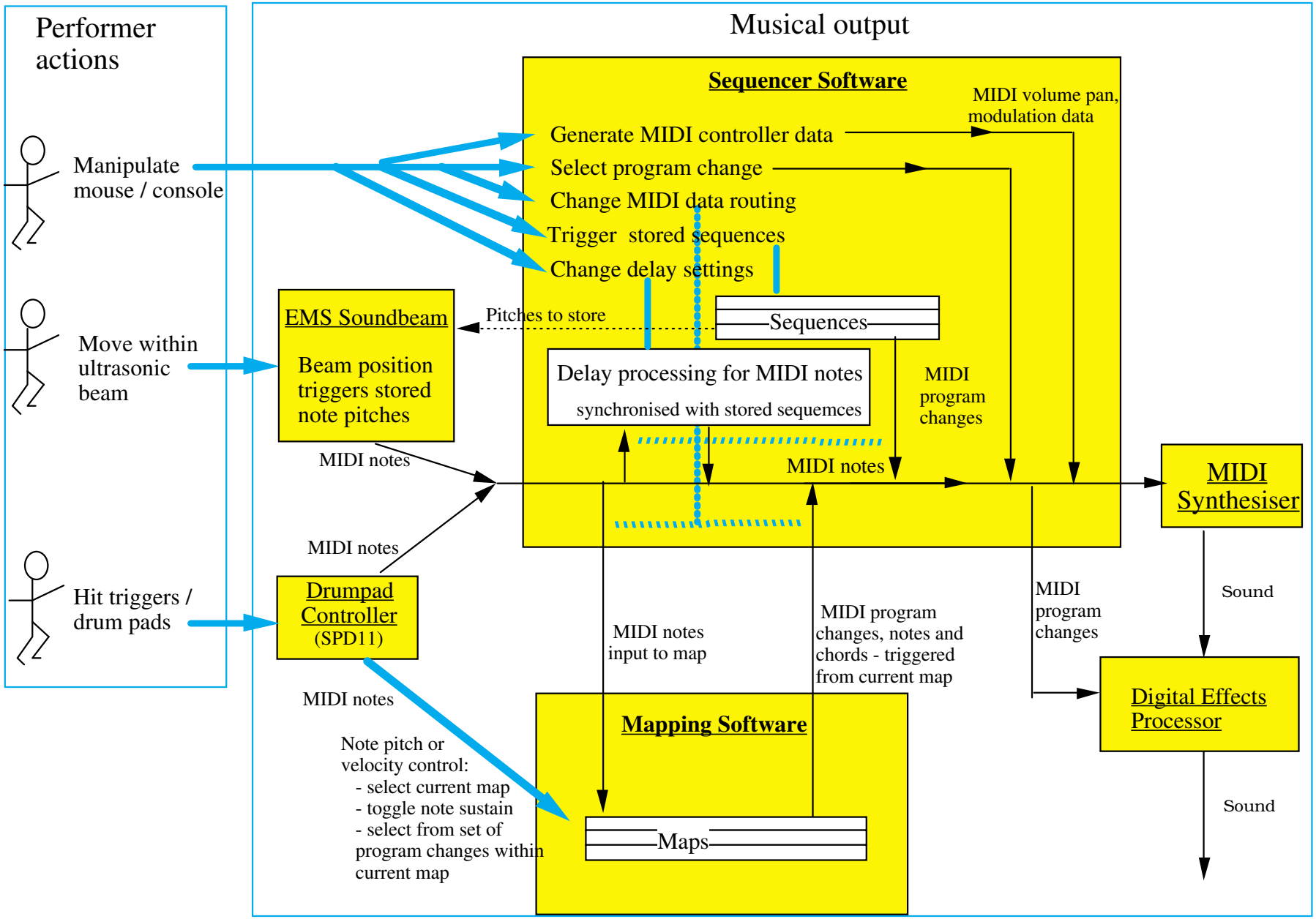


Hyper-instrument Example 4 Using an ultrasonic beam, pads and transducers to perform Pachelbel' s canon.

A distributed Hyper-instrument, using hardware and two software systems.

The emphasis is more on musical rather than on performer movement processing.




Performer actions
Manipulate mouse / console
Move within ultrasonic beam
Hit triggers / drum pads
Musical output
Sequencer Software
MIDI volume pan, modulation data
Generate MIDI controller data
Select program change
Change MIDI data routing
Trigger stored sequences
Change delay settings
EMS Soundbeam
Beam position triggers stored note pitches
Pitches to store
Sequences
Delay processing for MIDI notes
synchronised with stored sequemces
MIDI program changes
MIDI notes
MIDI notes
MIDI Synthesiser
MIDI notes
Drumpad Controller (SPD11)
MIDI notes input to map
MIDI program changes, notes and chords - triggered from current map
MIDI program changes
Sound
Digital Effects Processor
MIDI notes
Note pitch or velocity control:
- select current map
- toggle note sustain
- select from set of program changes within current map
Mapping Software
Maps
Sound