

Anhang

Anhang A - Abbildungen

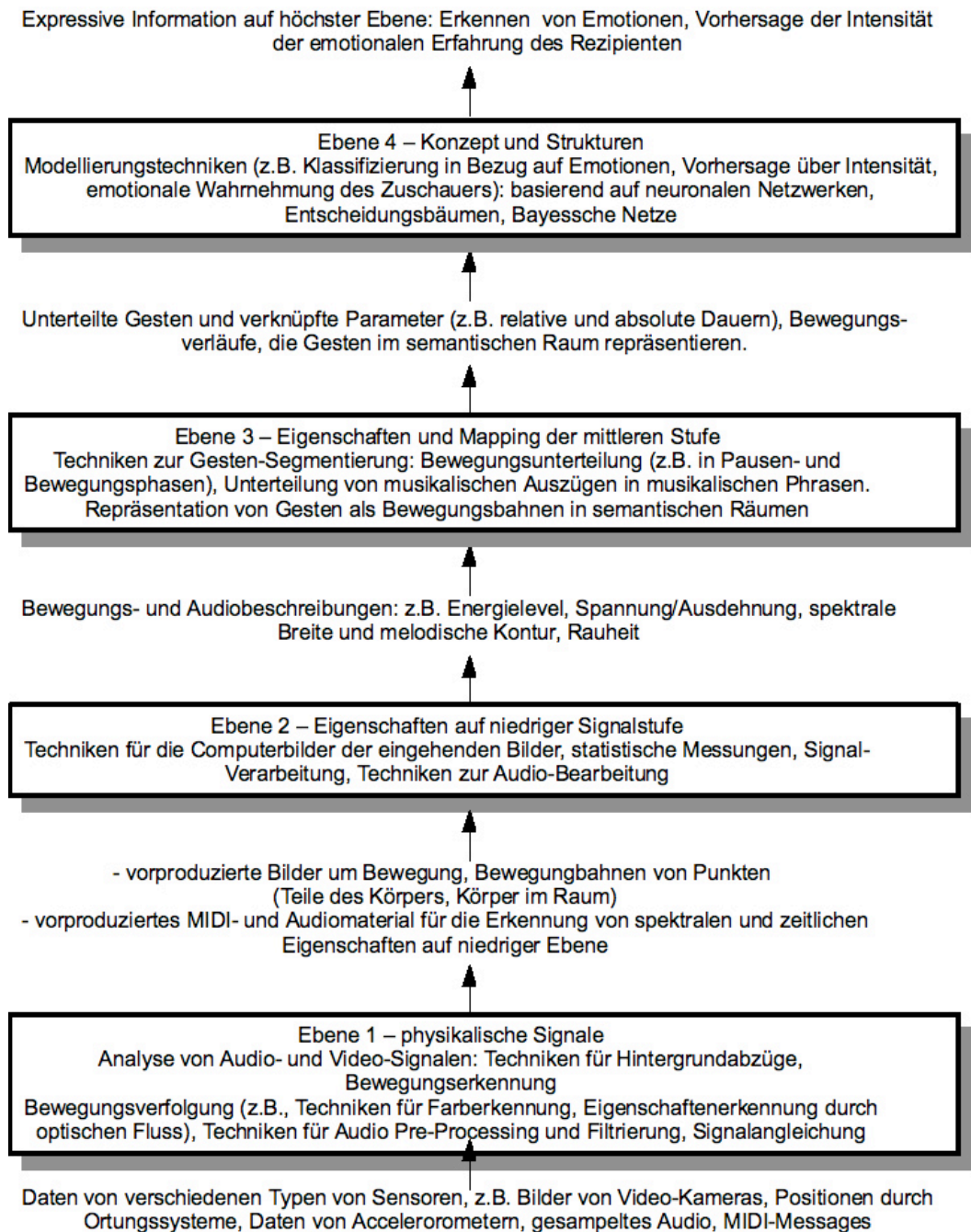


Abbildung 1: Schematische Übersicht des MEGA Project

Darstellung nach: Camurri, Antonio et al., *Modelling and analysing expressive gesture in multimodal systems* in *Paradoxes of Interactivity* hrsg. von Uwe Seifert et al. 2008, S. 223

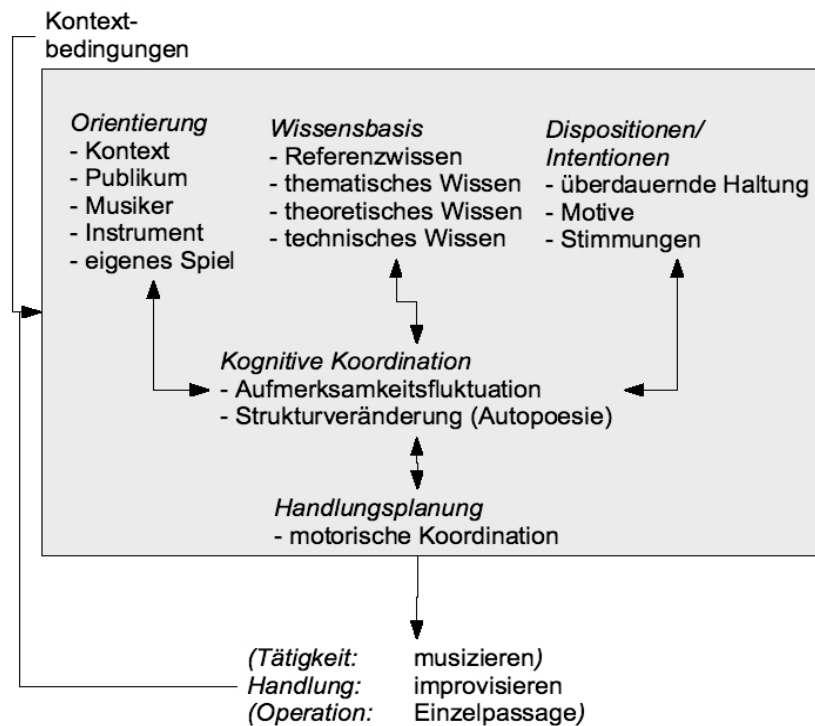


Abbildung 2: Handlungstheoretisches Modell der Improvisation

Modell entworfen nach dem Text in
 Andreas, Reinhard, Musikpsychologie, Musik der Geschichte und Gegenwart,
 Kassel, 1996, Sp. 599

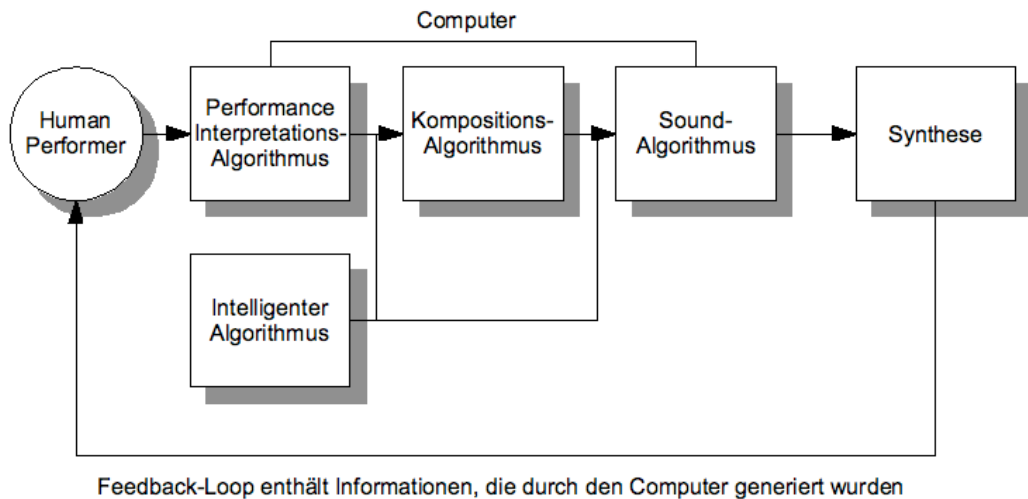


Abbildung 3: Schematische Darstellung eines interaktiven Musiksystems
 Schematische Darstellung zeigt den Ablauf des Systems wie beschrieben bei
 Chadabe, Joel (1983), nach Kim, J., *From Interactive Live Electronic Music to
 New Media Art*, in: *Paradoxes of Interactivity*, hrsg. von Seifert, U., 2008, S. 287

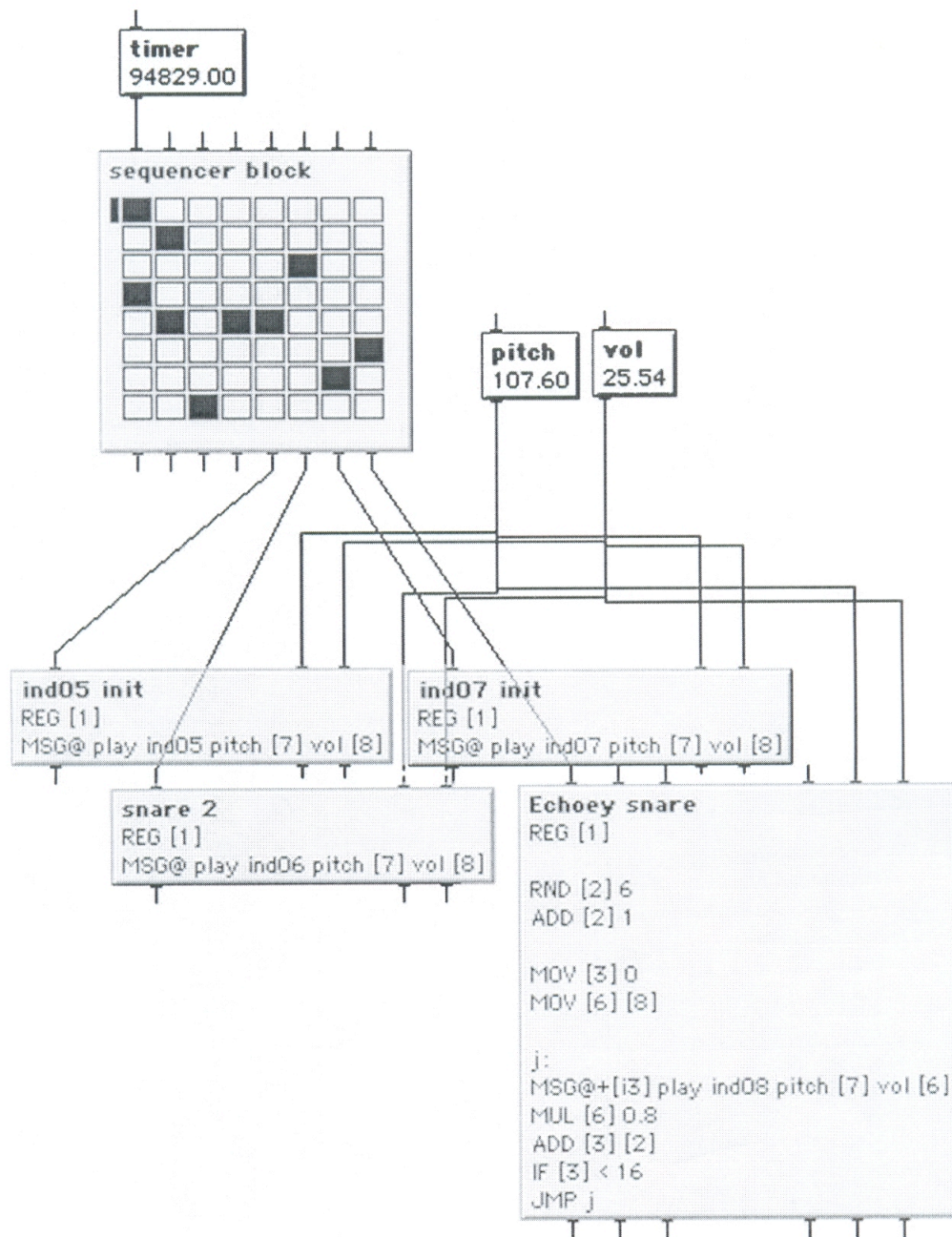


Abbildung 4: Maps von Adrian Ward

Maps ist eine Patcher-basierte Umgebung, in der zur Laufzeit nicht nur die Kabel zwischen den Objekten umgesetzt werden können, was Änderungen im globalen Programmfluss bewirkt, sondern auch die an Assembler angelehnten Befehle für die Struktur-Details.

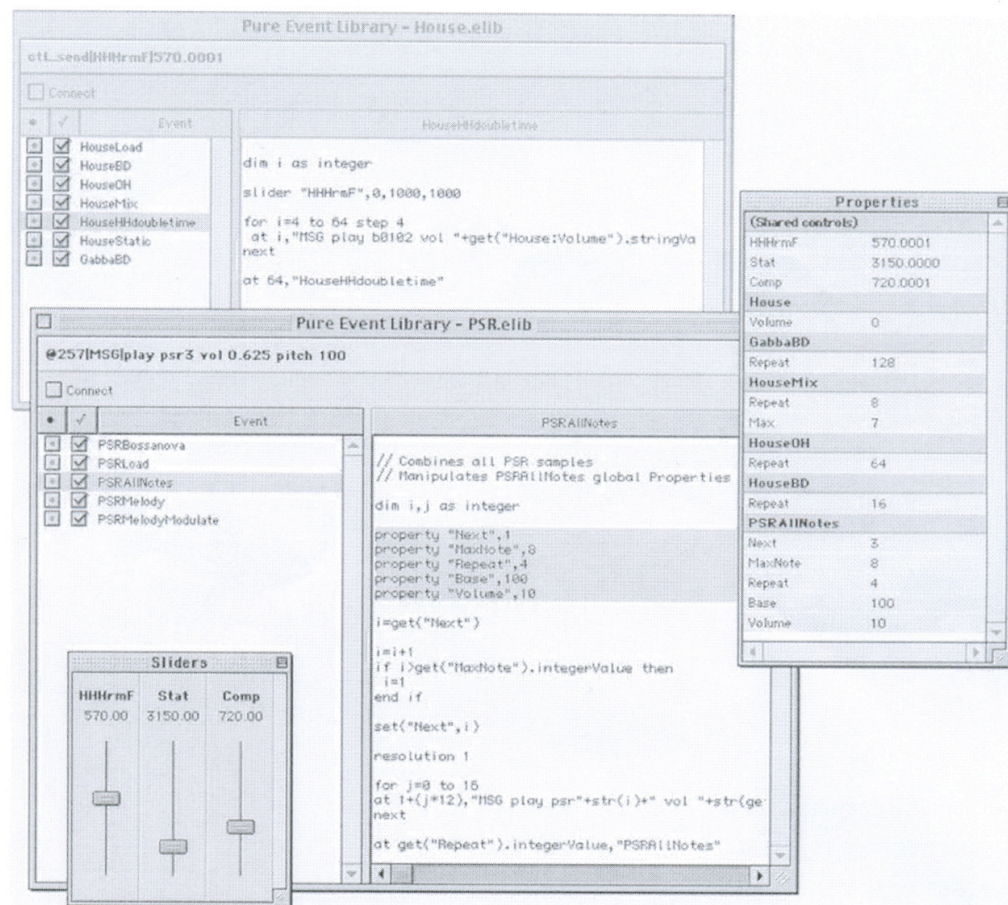


Abbildung 5: Pure Events von Adrian Ward

Snapshot der Performance-Software Pure Events, die die Live-Coding Strategie unterstützt. Routinen aus den Libraries werden aufgerufen und mit der GUI über den Property-Inspector verknüpft. Der Code der Module kann in Real-Zeit umgeschrieben werden.

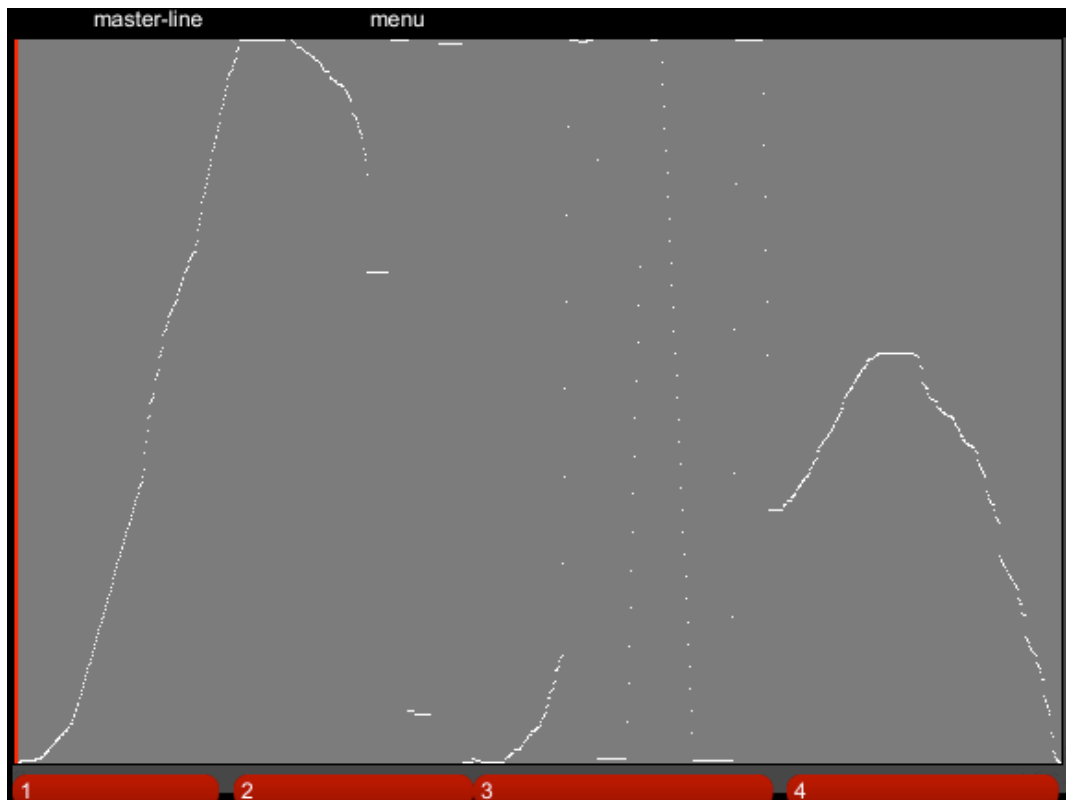


Abbildung 6: Master-Geste von 'Kurbelrichter'
Die roten Tabs zeigen die Abschnitte des Stücks

Lautsprecher-Konfiguration von Kurbelrichter

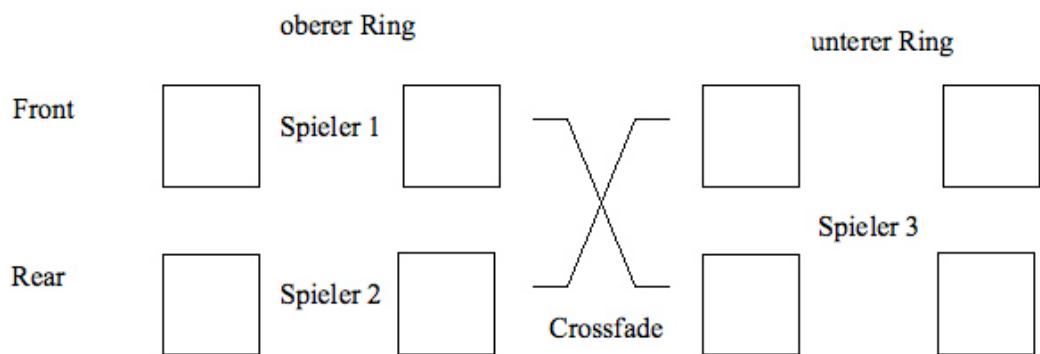


Abbildung 7: Lautsprecherkonfiguration bei Kurbelrichter

Der Crossfade deutet die Raum- und Effekt-Modulation in den späteren Teilen des Stückes an.

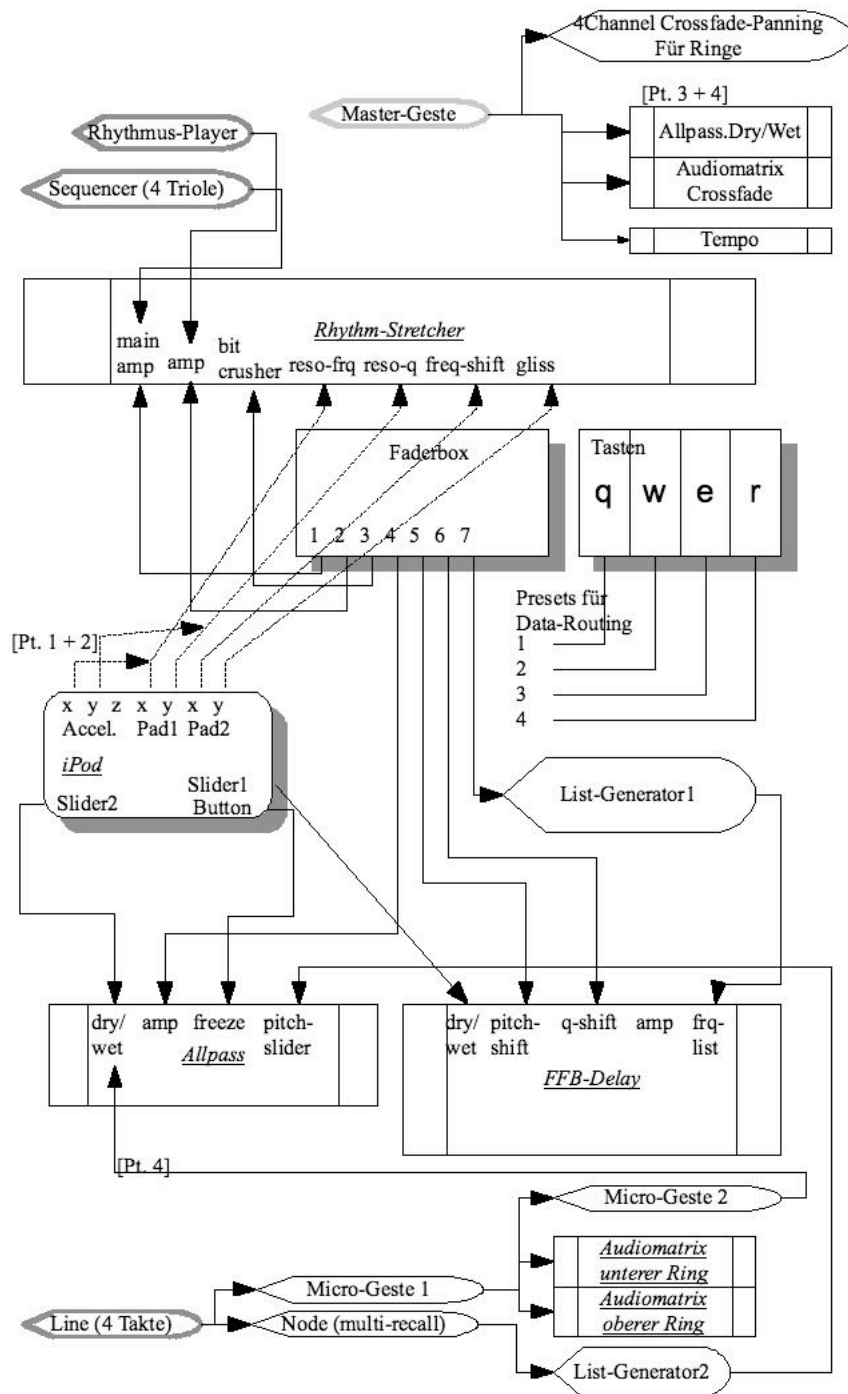


Abbildung 8: Schaltplan Spieler 3 Kurbelrichter

Es sind drei rechteckige Objekte zu sehen, die an den Seiten Säulen haben. Das sind die Audiogeneratoren (Allpass, FFB-Delay, Rhythm-Stretcher) mit den Parametern. Die Audiomatrizen gehören ebenso zu dieser Kategorie, werden aber

kleiner gezeichnet, weil sie nur Signale routen. Die ampullenartigen Objekte sind Strukturgeneratoren, die, wenn sie nicht gesteuert werden, über den Scheduler Tempo-basiert Ereignisse abhängig von den gesetzten Dauernwerten ausgeben (Objekte mit stärkerem Rahmen). In der Mitte der Skizze befinden sich die Interfaces – iPod, Faderbox und Tastatur – die entweder direkt Werte der Audiogeneratoren umsetzen oder auf die Strukturgeneratoren zugreifen. Oben rechts ist die Mastergeste, die Panning, Dry/Wet und Tempo steuert.

Anhang B

Traktat von Cornelius Cadrew zur Ethik von Improvisation

Vgl. http://ubu.clc.wvu.edu/papers/cadrew_ethics.html (Zugriff am 11.05.2009)

Virtues that a musician can develop

1. *Simplicity* Where everything becomes simple is the most desirable place to be. But, like Wittgenstein and his 'harmless contradiction', you have to remember how you got there. The simplicity must contain the memory of how hard it was to achieve. (The relevant Wittgenstein quotation is from the posthumously published 'Remarks on the Foundations of Mathematics': "The pernicious thing is not, to produce a contradiction in the region where neither the consistent nor the contradictory proposition has any kind of work to do; no, what *is* pernicious is: not to know how one reached the place where contradiction no longer does any harm".)

In 1957 when I left The Royal Academy of Music in London complex compositional techniques were considered indispensable. I acquired some -and still carry them around like an infection that I am perpetually desirous of curing. Sometimes the temptation occurs to me that if I were to infect my students with it I would at last be free of it myself.

2. *Integrity* What we *do* in the actual event is important -not only what we have in mind. Often what we do is what tells us what we have in mind.

The difference between making the sound and being the sound. The professional musician *makes* the sounds (in full knowledge of them as they are external to him); AMM *is* their sounds (as ignorant of them as one is about one's own nature).

3. *Selflessness* To do something constructive you have to look beyond yourself. The entire world is your sphere if your vision can encompass it. Self-expression lapses

too easily into mere documentation -'I record that this is how I feel'. You should not be concerned with yourself beyond arranging a mode of life that makes it possible to remain on the line, balanced. Then you can work, look out beyond yourself. Firm foundations make it possible to leave the ground.

4. *Forbearance* Improvising in a group you have to accept not only the frailties of your fellow musicians, but also your own. Overcoming your instinctual revulsion against whatever is out of tune (in the broadest sense).

5. *Preparedness* for no matter what eventuality (Cage's phrase) or simply *Awakeness*. I can best illustrate this with a special case of clairvoyant prediction. The trouble with clairvoyant prediction is that you can be absolutely convinced that *one* of two alternatives is going to happen, and then suddenly you are equally convinced of the other. In time this oscillation accelerates until the two states merge in a blur. Then all you can say is: I am convinced that either p or not-p, that either she will come or she won't, or whatever the case is about. Of course there is an immense difference between simply being aware that something might or might not occur, and a *clairvoyant conviction* that it will or won't occur. No practical difference but a great difference in feeling. A great intensity in your anticipation of this or that outcome. So it is with improvisation. "He who is ever looking for the breaking of a light he knows not whence about him, notes with a strange headfulness the faintest paleness of the sky" (Walter Pater). This constitutes awakeness.

Anhang C

Code-Beispiel für Live-Coding mit jitLib für Supercollider

Vgl. Collins, N. et al, Live coding in laptop performance, S. 328 f.

```
s=Server.local;
s.boot;
ProxySpace.push(s); // jitlib Initialisierung, Proxy-System
~out.play;

//~-Objekte sind Proxy-Bus Objekte, die über play hörbar gemacht werden,
//aber schon im
//Hintergrund mitlaufen

//klassische Sinuswelle
~out = {SinOsc.ar(440,0,0.05)};
// zeit für den crossfade
~out.fadeTime = 2.0

//die Neuzuweisung des Sounds wird durch einen Crossfade reguliert
//erstze diesen Sound durch einen anderen
~out = {LFPAr.ar(434,0,0.02)};

//erzeuge zwei k-Rates mit unterschiedlichen Sinusfunktionen
~x1={SinOsc.kr(0.1)};
~x2={SinOsc.kr(0.14)};

//kontrolliere die Rate von gefilterten Impuls durch x1 und x2
~trig = {Impulse.ar(30 *
    [~x1.kr.max(0), ~x2.kr.max(0)],
    0,2)
};

// teste
~trig.play;
~trig.stop;

~out= {RLPF.ar(~trig.ar(2),[400,410], 0.05)};
//Klasse von Alberto de Campo, um alle Busse, Instrumente und deren
Parameter
//anzuzeigen und zu verändern
ProxyMixer(p);

//x1 und x2 werden mit Sinuskurven moduliert
~x1 = {Mix(
    SinOsc.kr(2*[1,1.1,1.3,1.04] * ~x2.kr(1))
    ) + 1 * 0.2};
~x2 = {Mix(
    SinOsc.kr(2*[1,1.1,1.31,1.2] * ~x1.kr(1))
    ) + 1 * 0.2};
```

```

//neue Referenz, um Sound-Prozess hinzuzufügen
(
~out1 = {
    var amp, t;
    t = ~trig.ar(1) * 2;
    amp = Decay2.ar(t, 0.001, 0.01);
    BPF.ar(t, [400,431]*(amp+1),0.05)
};
)
//wirf den Kuehlschrank an
~out2 = {SinOsc.ar(4*50*[1,1.1], 0, 0.02)};

//mixe zusammen, Pulse werden durch out1 + out2 moduliert
~out = ~out1 + ~out2;

//bestimme Crossfade Zeit
~out2.fadeTime = 2.0

//Phasenmodulation des Kuehlschranks
~out2 = {SinOsc.ar
    (4*50*[1,1.03], SinOsc.ar(300,pi), 0.02)
};

//ein paar weitere Modulationen
~out2={SinOsc.ar(4*50*[1,1.032]*[1,~x2.kr*2+1],
SinOsc.ar(20) * pi * ~x1.kr, 0.03)};

//derselbe Modulator mit ein paar veränderbaren Parametern
~out2={arg freq=200, ffreq=20;
SinOsc.ar(freq*[1,1.03],SinOsc.ar(ffreq, pi) * ~x1.kr * pi,
0.02)
};

//setze die Parameter um
~out2.set(\freq, 240, \ffreq, 1200);

// verknuepfe Parameter mit k-Rate Variable
~out2.map(\ffreq, ~x1);

// aendere diesen Proxy
~x1 = {LFNoise0.kr(3)*10};

// eine neue Funktion benutzt die zuvor geaenderten Argumente
~out2={arg freq=200, ffreq=20;
SinOsc.ar(freq*[1,1.05],LFPulse.ar(ffreq * ~x1.ar(1),0,
0.5) * pi, 0.02)
};

```

Anhang D

Interview mit Matthew Ostrowski (O) geführt von Johannes Schmidt (S) am 04.05.2009.

Zu Beginn des Interviews geht es um das Live-Duo KRK von Ostrowski zusammen mit dem Kontrabassisten George Cremaschi

S: The pieces we are talking about are completely improvised?

O: They're quasi-improvised, in that they're pieces that - I mean-George and I have usually figured out a basic structure for most of the stuff. This is stuff that had been kind of worked out on tour over the course of the last year or so. So most of those pieces we both had a general idea of the sound worlds we were going to inhabit and the overall structure of the piece.

S: Can you say something about the technical setup?

O: Well, the technical setup is basically a slight variant of my standard improv setup. Basically I'm using the Max-Patch I wrote -- that's obviously including all the noise making stuff -- plus I'm using a P5 glove, which is a video game controller, which is actually doing most of the playing. And then in my left hand I have another video game controller which is basically 8 buttons and a scroll wheel and I simply use that to switch between my four main instruments.

S: What's the scroll wheel called?

O: Its called a *nostromo*. No idea what it was originally designed for, but its basically 8 buttons and a wheel and a rocker switch and that's it.

S: Both are HIDs?

O: Well, the *nostromo* is HID and the P-5 is a little weird. It doesn't use the standard HID and some guy in Australia, I'm sure you can find it online after two minutes, actually wrote a little UNIX program that grabs whatever serial data is coming out of the glove and turns it into open sound control messages. And that a little program is called P-5osc.

S: It's wrapped as an external in Max?

O: It's literally a stand alone UNIX program and it sends OSC messages on some port or another and you just grab that port locally. That's how it works. What that's giving you is XYZ position data and 4 finger bends. Well there's up and down, left, right back and forth, right?

S: It has a 7 bit resolution like MIDI?

O: No, no, no. It's a pretty high resolution. I've never tested it but it's high. The finger bends, however, are only 6 bit. The finger bends suck -- but that's a whole other technical problem I'm having.

S: On a picture I saw a little plastic tower in front of your setup. What's that?

O: The glove is infra red and that's essentially what's determining the position of the glove. It needs a reference point. I have to say visually I hate it but I can't live without it.

S: What is the advantage of an interface in your setup?

O: First of all, let me say that I actually started having to work with interfaces (when I switched from analog to digital in the early 90's) I've been through a lot of different things. I've been through a million different interfaces. Including fader boxes, keyboards, multi touch pads.... and for composed pieces I am still likely to go back to a fader box.

In terms of my code one thing that is sort of an aesthetic goal -- which obviously has to be manifested in code cause that's the only place you can manifest it -- the reason I like the glove and seem to have pretty much settled on it -- is what it allows me to do is to think of things that are happening musically not as a set of parameters that I can manipulate as individual things but rather as a single gesture. And in the course of those gestures I'm manipulating a lot of parameters.

But the example I always use is again string instruments. If I move the bow closer to the bridge I am changing several parameters. Or if I change the pressure of the bow I am changing several parameters with one gesture. Because an acoustic instrument is an interdependent system and so I try to write code that is

interdependent systems rather than simply mapping - I do some of that cause I'm lazy - but my goal is to have everything as interdependent as possible and not have my glove be a series of knobs but have it one more level of abstraction.

S: Instead of the glove you also could use another interface?

O: Yeah, you could do that with a fader box as well, but I think it's harder to visualize and to grasp what you're doing when you're looking at the positions of a row of faders. Even though you can theoretically move several at once. It doesn't hold in my mind as: 'If I move down across and move these 2 fingers that gives me something,' so I just feel like it's less playable in the sense that a real musical instrument is playable. Well, certainly from a theatrical standpoint it's got its advantages as well.

I have a mixed feeling on that issue in a certain way. Because having done stuff that was not very visible to the audience for years and years before I started using this glove - and obviously when I was doing analog synthesis that was also not very visible to the audience - I always felt that as a performer it is possible to maintain focus on you, as a performer, whatever you are doing if you are simply conscious of that and attempt to do it. I mean, even when I was using a bunch of fader boxes I never sat down.

Once I started using digital stuff I always tried to design it so that I had to look at my screen as little as possible. I always did think that regardless of what I was doing, the audience was going to see I was as present on stage as possible, and I think that's something people don't think about, but that has a lot to do with the current fashion of electronic music -- on which I have endless opinions -- but that's another issue.

S: Do you add any reverbs to the mix? Because the track sounds like there is an overall reverb.

O: Yeah, well, that would be in the house. We're not doing anything on stage, there's no pre-mixing on stage. There is occasionally processing in real time but other than that we're usually treated as two separate instruments. He also uses a

bass amp on stage. Which he may or may not use depending on the size on the house.

S: What kind of material are you using during a performance?

O: One of the things that we're interested in doing from an aesthetics standpoint is... Because his extended technique is very good, so we're trying to find that space where we overlap and integrate with each other in some way. So obviously if I'm processing him in real time then we're sharing an identical body of material. I have a fair number of string instrument samples (which may or may not be contrabass) that I also use. For some things, frankly, from a practical standpoint, I find it easier to use another contrabass sample with him rather than to feel I have a moral obligation to do everything that is supposed to sound anything like him in real time.

S: Why aren't you using the live-input from the contrabass?

O: It's an aesthetic decision in the sense of... in a typical -- I mean we can get into the glory details of what exactly my instruments are doing later on -- in a typical live processing situation the solutions that you hear 90% of the time are pretty obvious. What does someone normally do? Someone grabs a little chunk of it and uses it, loops it, transpose it and blablablabla.

So, what I've found just works more aesthetically is if I have stuff that's in his timbre universe, which is basically acoustic string instrument material. I can put it through any of my instruments which do a bunch of crazy shit which achieves the same general effect of his sound range, plus a computer manipulated version of his sound range, without necessarily being limited by whatever instruments I'm using that actually grab stuff in real time.

The thing that is important to emphasize here is... Us really occupying a similar timbre space is a relatively small part of what we do. But it's an axis of contrast and an axis of possibilities and I want to be able to go to that end of the axis.

So I can go 100% literal: do complete live processing and you're hearing almost identical material from the two of us. Or I can use my preexisting string sample

collection, and then you will hear something where we're roughly in the same department, or I can use stuff which is more and more contrasting with what he is doing -- and then the gap between us from an instrumental standpoint can become larger and larger. I think of the live processing as one end of an axis of similarity between me and George in terms of our general sonic space.

S: Why not use an algorithm for processing?

O: I don't consider that at all the emphasis of what we do. As a musician I have no interest in automatic things like that. Because I will be able to analyze and process and respond to something he is doing way better than any algorithm can.

I'm a human being and I'm smarter than a computer.

What I would certainly say as a general theory point is that my primary interest, my number one thing that I care about is the musical result. So something like that I feel is more of an exercise and I think also those kind of approaches, I feel, are better suited to non-realtime work.

Because I say in real time I think human beings are gonna kick the ass of that kind of stuff. And I've heard many voyager gigs. It's George Lewis -- one of the best musicians of all times -- and he takes years working on an algorithm and what does he come up with? He comes up with an algorithm and its okay but its sort of sad that he spends years working on an algorithm and its okay - but only okay.

I think all of those things are totally good for exercise to understand stuff and stuff like that.

S: What interaction rules do you implement in your system?

O: Before I get to that, you have to get the philosophical part and then we can talk about the geeky stuff which is: when I started doing live music there were basically two other people in NY actually doing live electronics. So my upbringing as an improviser was with primarily with acoustic musicians. Also, it was the 1980s a period of extremely fast changing, very active uh...

Yeah, it was that kind of hyperactive Zorn style, which he did not invent but everyone thinks of as being his style. So it was very important to be able to do

that. I couldn't just sit around and make textures which is what everyone does now, basically.

And one thing about an acoustic system – and this is the cross fade into the geeky stuff - is that acoustic instruments are energy in / energy out systems. If I do nothing to my saxophone it doesn't make any sound. Whereas computers have another power source, called electricity. So I don't have to do anything and the computer will still make sound. Back in the old days, when I was the only person who could make sound without putting energy in, I would set stuff up -- like doing all kinds of gags -- like setting something up and going out to buy a beer and then come back during shows. But that obviously... that was an interesting trick in 1985, but not anymore. So most of my software systems are designed that if I do nothing, nothing happens.

So that's sort of an important thing for me to think about. I have to put some kind of energy into the system in order for the system to make a sound. That's part of the general idea.

Do you actually want me to totally geek out on my various -- cause I usually have 4 different instruments that I usually use in an improv situations-do you want me to geek out on those and tell you what they are?

S: Perhaps we start with one of your instruments?

O: Although I have certain issues with it, the one that accomplished those tasks the best is a quasi-granular thing. What I have essentially done is: imagine you're looking at a sound file and what I've done is...there's kind of a virtual object which looks like a piece of truss... like a 2D object with an internal structure that holds it - so two parallel things with a bunch of cross bars.

But essentially what I'm doing is: I have a virtual object that is essentially traversing the space of a sound file, and the object is a Mass-Spring-Damper system. What it's doing is essentially obeying the laws of Newtonian physics.

S: What's the Mass-Spring-Damper?

O: Mass-Spring-Damper is... Imagine I have a weight, and a spring attached to another weight, and a spring attached to another weight. That's a Mass-Spring-

Damper system. And essentially it's this object which will obey the laws of physics which, with my glove -- I am essentially dragging it around by the nose and the rest of the chain is basically following my hand.

Every mass in that system is essentially identifying the point in the sound file where a grain is playing. So for example, if my hand is all the way to the left I'm at the beginning of the sound file. If I move it very quickly to the right, the whole system stretches, which means I am then getting samples from spread apart places in the sound file, and when I stop moving my hand the whole thing gets shorter because it wants to reach its position of rest. Then I'm dealing with a cluster which is much closer together. So if you think of your sort of typical granular thing where you have this parameter called window size, which is how big an area I'm picking my grains from. Normally that's an extremely linear thing. It's either one position or it's within a 10ms range or a 500ms range or a 1000ms range but it's always one area. Then I can get stuff from all -- a whole bunch of different places all at once.

Things like the velocity, the acceleration and those kinds of parameters are also controlling things like pitch and timbre and pan position and stuff like that. So basically the more active I am the higher the pitch, the crinklier the timbre, the more dispersed it is in space etc. etc. So that's the kind of idea where I'm doing a single gesture and there's this whole physical model layer between me and that set of parameters. So I don't have tons of direct parameter control. I have semi-direct control over pitch -- the y-axis is pitch -- but again, that will be effected by how quickly I'm moving as well. Well, let's say that's one of four instruments I use.

S: Besides the instruments do you use any effects?

O: Yeah, I'm not really into... well there is reverb, but the reverb is also controlled by these parameters. But generally speaking, I'm not very big on what one would think of as an effects layer. I think it homogenizes the sound too much, and most of the effects are sounds that we've heard a million times. We all know what a comb filter sounds like for God's sake. So I don't use an effects layer. Well certainly not when I'm improvising.

S: Why don't you use algorithms for structure generation or gesture recognition?

O: I would only be using algorithms on the structural level in pre-composed work. In improvised work no... My general rule -- well not my rule my habit, I haven't made some abstract decision this is just how it worked out -- is that I use algorithms for a lot of the micro decisions. But as an improviser I handle the macro decisions. And I will also say that as.... I fiddled around with a lot of... I have made a few attempts at algorithm gesture generation things, and I haven't really been happy with the result of any of that though. So I've kind of lost interest in that as a research approach, because I wasn't liking anything I was coming up with. So on let's say the two to five second level I try to avoid algorithm decision. It's mostly because I found it... it's really a matter of what I'm finding aesthetically effective or not. Some of my older pieces... I used to use the RTC toolbox a lot.

S: What's that?

O: RTC: the realtime composition toolbox. Its basically a bunch of Max-Patches to do twelve tone music. And I actually got some results I was very happy with using that and again in some of my more composed pieces I will very often use those things still. I guess what I'm saying is: I'm not opposed to it in principle -- as in if I get results that work I use it, and if I feel get results that doesn't work I don't use it.

S: Dahlhaus says improvisation is only a recall of learned patterns and always needs a musical structure to improvise over, like in jazz, because major decisions like form can never be improvised.

O: Well I think that's useless dogmatism. Well, first of all, that is not true of jazz. That is true of bebop. Go listen to some dixieland where everyone is soloing at the same time. And then go listen to something like Ornette Coleman, where everyone is soloing at the same time 70 years later.

So that whole like trading fours blablabla, that was certainly a very significant and long phase of jazz but that is not what jazz is. Well, that is kind of a variant of Cage's complaint about jazz which was that everybody just falls back on their

habits. But on the other hand Morton Feldman, he just falls back on his habits. Right? It's like you put the needle down on the record and you know it's a Feldman piece in 30 seconds. And nobody complains about that. Or for that matter Mozart. I mean, that's what's called style. And I don't have a problem with style per se. My problem with improvisation is not theoretical but again practical which is that most people especially, sorry to say it, the kids nowadays don't understand that improvisation is in fact a very difficult discipline that requires a great deal of practice and a great deal of playing, and is not just like: oh, I'm just going to do what I feel like right now. And it's weird 'cause I feel a lot of the criticisms that were leveled at free music back in the '70s are more true now than they were then, precisely because a lot of younger musicians have completely lost any kind of connection with what is the original source really of improvised music, which is jazz. And I don't sit around and listen to jazz all the time but I have and I understand... It's like everything else: you've got to listen to stuff you've got to develop an ear, you've got to learn what's good and what's bad by listening to stuff. But the sort of jazz idea is this is work, this is discipline, this is hard.

Also, I designed my instruments so they are nonlinear. I don't know what they're going to do really which means the only way I can get familiar with them is by playing them.

It's funny -- I was just at the Max-Expo, and a couple people were showing their stuff, and what you see is you wind up with people who basically have some setup where you can control every imaginable parameter of whatever it is. And then they're thinking of what they're playing as a bunch of parameters they're not thinking of what they're playing as an instrument. All of my little algorithms are pretty limited, but they're nonlinear which means -- okay, there's only a certain set of sounds I can get out of them. But what is my job as a musician to do is to find out how to get interesting sounds out of it and how to structure those sounds.

S: But non-linear systems are hard to control and to realise as a precise idea.

O: Well, that depends on what you mean by precise. I mean, again a saxophone is nonlinear. And people can control the saxophone pretty well.

What distinguishes the good improvisers from the bad improvisers is the ability to spontaneously develop a structure in real time. I make a conscious effort to vary that structure so lets say I'm doing a totally cold gig with a guy I played with once 6 years ago. So what am I going to do? Well I'm going to start some pieces with a bang really suddenly with a big gesture. I'm going to start some pieces really slow moving in and see how things go. Again this is just a matter of experience and concentration. But I am very aware of the fact that 90% of the crap you hear -- especially on laptops -- is a goddamn arch form. You know you start really quiet and you get really loud and you get really quiet again at the end.

This is the thing that's really interesting about improvised music. What you want to be able to do is create some kind of, not necessarily a symmetrical structure, but some kind of developmental structure that... Okay, I'll tell you a good story:

A couple months ago I did a gig , actually it was with George, and my mother's husband, who I think has seen some of my installations, but he had never seen me do a free improv gig -- he's a big giant classical music head and he came to the show, and after the show he said: I could recognize everything. Although the material you were working with was very different, I could hear themes, I could hear variations on themes, I could hear scherzos -- you know I could hear all the stuff I can hear.

So it's the same rules of music. It's stability vs. variation, predictability vs. unpredictability, large size gestured vs small size gestures. This is true no matter whatever kind of music you're doing.

And the thing about improvisation is you just have to learn how to do it. Which means doing it a lot, listening a lot to what you've played. That kind of stuff. Its just practice! Again practice! Practice, practice.

S: Are you trying to establish a new way of listening? Like creating new forms, bring people to do 'active reception'?

O: I don't know if you can... how to put it... Let's talk about Feldman again for a second. Obviously Feldman dials everything way down -- he does little things. But what he's doing is -- It's still the same thing, it's just on this much more micro

level. It's still -- how is this little cluster of tones (which you then have to wait for thirty seconds for the next cluster of tones) -- how does this differ from the previous one. Those things are always there. You can't avoid that and I think it's.. yeah... this is one reason that I'm interested in using more asymmetrical forms and things like that but I feel like there's a lot of: 'Oh I want people to get rid of their old listening habits so I'm going to subject them to a 120dB wall of noise for 45 minutes.'

And it's like... yeah.. Actually, Merzbow, compared to a lot of the other wall of noise shit I've seen is actually pretty good, because at least his wall of noise has some detail in it. But the fact is to do that kind of stuff is really really hard and even so, those things are also true.

I try doing things like that by avoiding as much as I can the concept of foreground and background. I'm interested in lots of little micro events and making people concentrate on the micro events and I'm interested in all those kinds of things which can also make people change their listening habits.

But ultimately lets put it his way I feel like that gets used as an excuse a lot to just do something dumb. Like the 45 minutes wall of noise and then you go 'well I'm changing peoples listening habits' and you just go: sorry, it's fucking boring.

S: How did you get to electronic music?

O: Well I got involved in electronic music by mistake when I was very young, and in those days you basically couldn't do live electronic music because you needed big giant refrigerator-size synthesizers and stuff. So when I was first doing this as an undergraduate, basically the only game in town was four-channel tape pieces. That's kind of what you did 'cause that's what there was. And as a student I would work on these pieces and I felt that - sort of critiquing my own work - I thought my sense of timing really sucked, and I detected a lot of weakness is in my own work obviously, but the main one was that I felt that my sense of timing really sucked.

And at that time I was listening to a lot of different kinds of music -- but I was also very much in touch with a lot of no wave stuff that was happening and a lot

of the improvised music that was happening in New York, and one of the things that attracted me so much to that music was the way that things unfolded in time. Because again I had very little traditional music training and I felt like the way for me to develop my sense of time and my sense... 'cause really, what's music about? Its manipulating time. And I felt the way for me to develop that was by doing improvised music. Because I heard all this improvised music that was doing all these amazing things with time. And I know that there's a bag of tricks that traditional composers used to manipulate time. But frankly, I flunked music theory and I really tried not to. I was just a total loser.

And I felt like: 'okay the tricks you learn in composition class are not readily available to me. So how do I develop those skills?' So I started doing improved music actually with the idea of taking those skills back into the realm of tape music (which I still do), but then of course you get more and more involved in live music, and then you find that you're primarily playing live. That's how I got involved in it basically. But I did not... Electronic music, I just stumbled upon it and got obsessed. That's pretty much that.

S: Matty, thanks a lot for this interview.

O: I love talking so happy to help. Talk to you soon, bye.

Anhang E

Interview mit Prof. Dr. Alberto De Campo (D) geführt von Johannes Schmidt (S)
am 24.05.2009.

S: Was an PowerBooks_UnPlugged ist tatsächlich Live-Coding?

D: Ja, wenn man so die Extrempositionen vom Live-Coding oder Just in Time - Programming diametral gegenüberstellen will, könnte man sagen, die Parodieversion von Live-Coding unter verschärften Bedingungen wäre, man muss mit dem leeren Bildschirm beginnen, darf gar nichts vorbereitet haben und muss alles von Scratch neu schreiben. Also, der Unterschied zwischen Live-Coding und diesem strengeren, der ist ja eigentlich auch so ein bisschen provokant, vergleichbar mit der Virtuosität von Instrumentalisten, zugespitzt. Da gibt's ja Leute, die das durchaus machen, die versuchen, den Begriff für die Richtung ein zu schärfen. Ich finde es interessant, über musikalische Ideen, die als Programme formuliert sind, nach zu denken, gleichgültig, wie viele es von denen schon vorher gibt, und von denen kontinuierliche Veränderungen, die für Zuhörer interessant sind, machen zu können.

Meine eigene Praxis sieht so aus, dass ich das am liebsten mit anderen Leuten zusammen mache. Ich finde eigentlich Gespräche auch interessanter als Monologe und habe nicht unbedingt den großen Ehrgeiz, das als Soloperformer vergleichbar mit einem Violin-Virtuosen machen zu können. Ich finde eigentlich Gruppensituationen dann interessanter.

S: Das bedeutet, dass der Austausch dann nicht mehr auf ästhetischer Ebene stattfindet, die man vorher diskutiert, sondern tatsächlich über den Code selbst, wenn man ihn performt. Ist es bei Ihren Performances so, dass man Code-Fragmente hin- und herschickt oder ist es so, dass ein Datenpool oder ein Code-Pool klar ist, den jeder Performer benutzt, oder gibt es da noch andere Möglichkeiten?

D: Grundsätzlich ist die Idee musikalische Gedanken im Gespräch weiter zu entwickeln, dadurch dass man den Text, den der andere gerade gemacht hat, zur

Verfügung bekommt, hat durchaus etwas damit zu tun, dass man jetzt z.B. als improvisierender Jazzmusiker ja auch hört, was die anderen machen, und z.B. neuere Improvisationen aufgreifen könnte oder in die Figuren, die ein anderer in einem Solo vorher gespielt hat, oder jemand in einer Begleitfigur spielt, auf zu greifen und weiter zu bearbeiten. Das ist ja im Prinzip von der musikalischen Praxis her dasselbe. Also ich gehe drauf ein, was die anderen Leute machen, weil ich es über entweder über das Hören kriege, oder auch in der Form, in der wir das machen. Mit PowerBooks_UnPlugged ist es technisch so, dass wir immer über zwei, eigentlich drei Ebenen mit einander verbunden sind. Also wir heißen zwar unplugged, aber das drahtlose Netzwerk gibt es ja trotzdem. Die drei Ebenen sind, wenn ich z.B. eine kleine Sound-Struktur programmiere, die einen Strom Sound-Events erzeugt, dann können die Sound-Events auf meinem Rechner passieren oder bei jemandem anderen, der in der gleichen Gruppe ist, der im gleichen Netzwerk ist und eigentlich auch einen Sound-Server laufen hat. Gleichzeitig geht der Text den ich ausführe an alle Mitspieler. Jeder kann lesen, was alle anderen dann gemacht haben. Es gibt eine gemeinsame History von der Performance, in die jeder zurückgehen kann, die auch jeder suchen kann, Dinge, die vorher passiert sind, wieder nehmen kann, also darauf zurück greifen kann und die verändern kann.

Und als dritte Ebene gibt es noch so vereinbarte Signale, die im Prinzip quer über den ganzen Bildschirm dargestellt werden, dass man andere Leute darauf aufmerksam machen kann, „ich bin in entsetzlichen Schwierigkeiten“ oder soll man jetzt zu dem und dem vereinbarten Formteil kommen oder sonstige Absprachen, die bei offenen Improvisationen sinnvoll sind. Das sind quasi so die drei Kommunikationsebenen, die wir zur Verfügung haben und auch alle dann verwenden.

S: Was sind so die Ziele, die vor der Performance so abgesprochen werden?

B: Wir überlegen die Sache meistens sehr kurzfristig vor der Performance. Und die passieren dann auch nicht unbedingt so, wie wir das abgesprochen haben, aber es gibt zumindest Absichten. Wir haben für das Konzert in Budapest auf einen

Zettel Papier einen Verlauf für die ganze halbe Stunde skizziert, die wir geplant haben zu spielen, in welchen verschiedenen Zonen von Sounds wir vorbeikommen wollen. Diesen Zettel fotografierten wir und haben ihn als Bildschirmhintergrund für jeden draufgespielt, damit man zwischendurch mal draufsehen kann, wo sind wir denn jetzt und wo wollen wir vielleicht denn noch vorbeikommen.

Im Prinzip ändert sich jede Performance-Situation je improvisierter es ist. Da ist sowieso jedes Mal wieder der Zusammenhang so, dass man jetzt nicht strikt den Plan unbedingt durchziehen muss, sondern dass es interessanter ist zu schauen, was passiert denn alles noch an Abzweigungen, bis man zur nächsten vereinbarten Sache hinkommt. Die vereinbarten Sachen, die wir machen, sind eigentlich nur sehr ungefähr definiert. Also so Dinge wie, dass wir beschliessen, in der Mitte sehr dünne Texturen zu machen, sehr viel Platz zu lassen, also die Strukturen auseinander fallen zu lassen, um dann eine Schlussrampe zu machen, die sehr noisig und sehr laut wird, so laut es mit Laptops geht.

S: Sie benutzen keine P.A., keine Verstärkung. Sie benutzen nur die kleinen Quäken von den Laptops, habe ich zumindest so gelesen. Habe ich das richtig verstanden ?

D: Bei einem Workshop mit Julian Rohrer in Köln mit ungefähr 10 Leuten wollten wir untersuchen, wie sich Granularsynthese in Netzwerken verhält, weil man da interessante Effekte über Laufzeiten über Verständigung in Netzwerken bekommen kann. Dann haben wir uns überlegt, wie wir das Lautsprecherset dafür machen. Alle auf Stereo zu mischen wäre eigentlich Schwachsinn, weil man dann gar keine Ahnung hat, wer wo welchen Sound spielt, wer Sounds hin- und herschickt. Jedem einen kleinen Lautsprecher oder einen irgendwie so normalen Studiomonitor-Lautsprecher zu geben, war dann auch ein bisschen umständlich und dann haben wir beschlossen, es mit den kleineren Lautsprechern zu versuchen. Das ist eigentlich superinteressant, weil die einerseits dann wirklich als physikalisches Instrument das mit dem Laptop machen, wo man auch die Größe von der Kiste hört und eben auch charakteristische Dinge wie „unter 250 Hertz

passiert nicht mehr viel“ oder interessantes Verzerrungsverhalten in schreienden Formanten. Je nachdem welcher Typ von Laptop ist zwischen drei und fünf kHz so ein Buckel, der recht kräftig nach vorne springt. Die Sachen kann man alle auch als Instrumentendesign auffassen oder als interessante Einschränkungen, mit denen man auch umgehen lernen kann. Ich finde, es hat eine sehr schöne Wirkung, wenn wir jetzt verteilt im Publikum sitzen und da Leute rund herum auch nahe bei uns sitzen und dann einem über die Schulter schauen und mitlesen können, was der oder die gerade so macht. Ich finde, das erzeugt einen sehr netten Kontext für die Performance.

Manche, also jetzt zugespitzt, wenn man so extrem laute Noise-Musiker z.B. die mit Laptop spielen, nimmt und sich da überlegt wie groß der Unterschied ist zwischen minimaler ausdrucksloser Gestik, ab zu mal ein Mouse-Klick, irgendwie ultrabrutale Rock-n-Roll-Lautstärke, dann liegt das ja irgendwie weit auseinander. Ich finde es auch interessant, auf ähnliche Art ein Gefühl für Intensität auf viel geringeren Lautstärken zu kriegen. Ich finde, es funktioniert nicht schlecht mit den Einschränkungen.

A: Reduzieren sie dadurch nicht die Vielfalt im Klang?

B: Die Sounds sind speziell dafür gemacht, dass sie in dem Setup interessant klingen und viele verschiedene Farben und viele verschiedene Bereiche abdecken können eigentlich. So leise sind Laptops gar nicht, weil man sich drauf eingestellt hat. Und mit allen Einschränkungen können die schon eine ganze Menge. Wenn man funky Basslinien machen will, kann man das auf Laptops auch machen. Da muss man schon so weit in die Bass-Verzerrung reingehen, dass man schöne virtuelle Grundtöne kriegt. Für den Zusammenhang von PowerBooks_Unplugged machen wir das schon so, dass wir in den allermeisten Fällen unverstärkt spielen, ab und zu nehmen wir auch Subwoofer hinzu, wenn sich die Gelegenheit dazu bietet. Wir haben auch schon mal Doppelkonzerte gespielt als PowerBooks_Unplugged – PowerPluggs Unbooked, wo die PowerPluggs dann schon mal verstärkt spielen und dann technisch in einem ähnlichen Stil mit netzwerkverbundenem Live-Coding operieren, aber halt andere Sachen spielen,

die über Lautsprecher auch gut klingen. Da verwenden wir meistens jedoch Lautsprecher-Setups, die so eine gemischte Form von Multikanal ist, dass man zumindest einiges von der Aufbereitung von den Klängen doch Verstärkung da ist. Das geht auch ganz schlecht. Das wächst dann so ein bisschen in ein paar andere Ensembles rüber, wo z.B. der Hannes Hoelzl und ich mit anderen improvisierenden Musikern, die akustische Instrumente spielen oder elektronische Instrumente, wo es mit denen zusammen total Spaß macht, die Sachen auch ein bisschen durcheinander zu mischen.

S: Ist es nur SuperCollider und wenn ja, warum?

D: In meinem Fall ist das allermeiste, was ich mache mit SuperCollider. Einfach aus dem Grund, weil ich die letzten – muss ich nachrechnen – ja mehr als zehn Jahre eigentlich – die allermeisten Dinge damit gemacht habe und mich in der Sprache auch wirklich wohl fühle und die meisten Sachen damit schneller machen kann als in jeder anderen Umgebung. Zum Teil gibt es auch Dinge, die man von der Sprache geschenkt bekommt, also, dass die Sprache so wie so strukturiert ist, also dass man ein Client hat, in dem die Strukturprogramme laufen und dann einen Server, der nur DSP macht, die über Netzwerke verbunden sind. Das macht total offensichtlich, dass man das ganz schnell auch über ein Netzwerk verbinden könnte.

Insofern kann man auch sagen, dass das ein Benutzen von dem natürlichen Idiom von der Sprache ist, dass sich in dieser Richtung entwickelt hat. Andererseits gibt es für SuperCollider den Grund, dass es die Erweiterungsbibliothek jitLib gibt, die Julian Rohrhuber entwickelt hat, und bei der ich ab und zu ein kleines bisschen mithilfe. Diese ist für just-in-time-Programming wirklich interessant und für mich ganz toll das Konzept umsetzt, dass man nur die Absichten hinschreibt und immer weniger Verwaltung drum herum programmieren muss, damit die Dinge auch wirklich passieren.

Das finde ich einfach vom Denkt tempo her angenehm und das macht auch so Dinge wie, eben dass man den Code von anderen Leuten liest und dass man versucht, darüber nach zu denken und daran weiter zu bauen. Das geht auch erst,

wenn man nicht irgendwie zeilenweise so Administration dazwischen klemmt, die man jetzt auch alle lesen müsste, wo man Live gar nicht die Zeit dafür hat.

S: Was sind so die Hauptmerkmale von jitLib, die Ihnen bei der Umsetzung des Live-Codings verhelfen?

D: Ein Teil von jitLib heißt Proxy-Space und ist eine sehr elegante Variante Synthese-Prozesse miteinander zu verbinden. Genauso wie ich meine *Tdefs* von vorher definierten Abläufen in der Zeit durch Neuabläufe, durch neue Tasks ersetzen kann, kann man in Proxy-Space Synthese-Prozesse durch neue Synthese-Prozesse mit dem gleichen Namen an der gleichen Stelle ersetzen. Das funktioniert alles mit eleganten Crossfades so, dass man ganz viele Strukturen auf Verdacht erst mal so langsam Teil für Teil bauen kann und die erst am Schluss zusammenstecken muss. Da hat man genau die Patchchords mit eingebauten Crossfades, die man schon geschenkt kriegt. Die sind immer knackfrei, außer man macht da wirklich Programmierfehler, wo schrecklich Lautes rauskommt. Aber sonst ist das absolut wohlerzogen und Livecode-fähig. Also auch kultiviert Livecode-fähig.

S: Verliert man bei all den Daten- und Audio-Bussen nicht schnell die Übersicht?

D: Das war mit ein Grund, die Proxymixer-Klasse zu schreiben. Im Prinzip müsste man eigentlich selber im Kopf behalten, was hat man denn jetzt von dem ganzen Zeug, das hier steht, schon tatsächlich ausgeführt und was ist der aktuelle Zustand von allem. Und in Konzerten fällt das relativ schwierig den Überblick zu bewahren. Außer ich mache nur irgendwie sehr überschaubare Dinge. Wenn wir fünf, sechs Dinge gleichzeitig haben oder so, wäre dann quasi so das Maximum, was ich so halbwegs im Kopf behalten kann.

Und da ist es dann einfach prinzipiell schwierig darauf zu kommen, wo man denn eigentlich jetzt gerade ist. Es ist ja auch nicht eine gerade Linie von allem, von den ganzen Dingen, die passieren könnten, sondern man kann ja auch überall hin springen. Und dann hat man jedes Mal einen aktuellen Zustand, den man zwar selber erzeugt hat, aber wo man sich vielleicht nicht an jedes Detail erinnern kann,

was man jetzt gerade schon ausgeführt hat und ab wo habe ich jetzt Out2 als Letztes, auf das eine Ding oder das andere Ding ausprobiert. Wenn das zwei Minuten her ist oder drei, dann kann ich aber selber nicht mehr sicher sein, welche von denen es jetzt gerade ist. Und dafür finde ich es ganz schön, so gerade so viel Feedback zu kriegen, dass das funktioniert, dass man den momentanen Zustand relativ schnell versteht und nachschauen kann. Ich kann auch, wenn ich einen Proxymixer dehne und jetzt nicht weiß, was war denn Out2 als Letztes, dann kann ich mit Out2 auf den Editor gehen und im Editor auf 'doc' drücken, dann sehe ich eigentlich alles, was in Out2 verwendet wird und das, was Out2 als Letztes, der Text in Out2 als Letztes erzeugt hat.

S: Wie überschauen Sie die Zeit beim Verändern des Codes in einer Performance?

D: Das ist ja ein prinzipielles Problem, die Idee live Code schreiben zu wollen. Da kann man leicht dran verzweifeln und meinen, es geht sowieso nicht. Die Lösungen, die wir dafür haben, sind einerseits dafür, vor dem Konzert den Pool von den ganzen Sachen, die wir in den früheren Konzerten oder Proben schön gefunden haben. Jeder kann darauf zugreifen und kleine Scripts von den Bandkollegen und -kolleginnen auch nehmen und spielen.

Also die geniale Patentlösung haben wir dafür auch noch nicht. Wenn ich mit akustischen Musikern spiele, kriege ich noch immer den gleichen Neid darauf, wie die in Sekundenbruchteilen ganz woanders hingehen können, und dann sofort dort sind und in einer ganz anderen Welt weiterspielen können. Ich weiß noch nicht, wie man das Tempo vernünftig am Computer kriegen soll. Man kann sich schon überlegen eine Riesensammlung von Pages, die man sofort am Start hat, also mit einem Knopfdruck parat liegen zu haben und dann sofort in ein spezielles Preset reinswitchen zu können, um dort weiter zu spielen. Das wäre durchaus so ähnlich wie improvisierende Musiker auch bestimmte Idiome draufhaben, die sie vorbereitet haben, wo sie wissen, wie sie die ersten paar Sekunden davon spielen und daran dann weitermachen. Also insofern ist das nicht groß was anderes, als genug Skalen oder Konzepte oder was immer man sonst an Material nimmt, geübt zu haben und insofern von der Spielpraxis parat zu haben.

S: Ich bin Ihnen sehr dankbar für das Interview, Herr De Campo.

Anhang F

Partitur Kurbelrichter

Master Geste	Rampe asc - I	Berg/Sprung – II
Typus	Tempo accel, Kurbel	Puls/Rausch
X-Achse (Zeit)		
Y-Achse	Tempo	0 – Puls 1 – Rauschen Tempo
P1	1. 4nt - > auf ganzen Tag ausweiten, Pause 1 +2 einleiten	Puls + Rausch im Wechsel
P2	2. 4nt, dto.	Puls + Rausch im Wechsel
P3	3. 4nt, dto.	Rauschfläche bleibt
Video	2D, Licht am Ende	Stufen, Wire, Warp

Master Geste	Sprung – III	Rampe desc - IV
Typus	Dry/Panning, Wet/Ring Puls	Dry/Ring, Wet/Panning, Puls, Fläche, ausfransen
X-Achse (Zeit)		
Y-Achse	0 – Dry, langsam Pan 1 – Wet, schnell Pan Tempo	
P1	Pan über Geste, Front	Wie I invertiert
P2	Pan über Geste, Rear	Wie I invertiert
P3	Stretch-Rhy Dry/Wet über 4T Geste, 4 Chnl	FfbDelay, Stretch-Fläche
Video	Perlin über 0/1 steuern	Cubes, Alpha, fadeout

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst und keine anderen Quellen und Hilfsmittel als die angegebenen benutzt habe. Die Stellen der Arbeit, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, habe ich in jedem einzelnen Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht. Dies gilt auch für die aufgeführten Notenbeispiele.

Essen, den 15. Juni 2009

Johannes Schmidt