

JetztMusik / Musik spezial

Wo die Quellen offen liegen

Freie Software und aktuelle Musik

Rundfunkmanuskript von Michael Iber

- Zuspiel 1: O-Ton Ivica Ico Bukvic: *Applications always influence artists and how they work. This is something I discussed with many other people who actually use extensively computer technology to produce electro-acoustic music. And it's unavoidable, you know: the interface itself lends itself to particular ways of manipulation. And I think in the case of Linux because of this diversity and the open source aspect of it, I think that flexibility is so much greater*
- Zuspiel 2: Ivica Ico Bukvic: Legisonitus I "Gone in 8 minutes and 11.527 seconds"
(Privataufnahme des Komponisten)
- Sprecher 2: [über Zuspiel] Grundgesetz der Bundesrepublik Deutschland Artikel 5, Absatz 3: Kunst und Wissenschaft, Forschung und Lehre sind frei.
- Zuspiel 2: pause
- Sprecher 1: Ein Haus können Sie kaufen und, wenn Sie wollen, können Sie es verändern. Sie können ein Musikinstrument kaufen und damit tun und lassen, was Ihnen gefällt. Auch einen Computer können Sie kaufen und nach Ihren Vorstellungen ausstatten und umbauen. - Software hingegen können Sie nicht kaufen. Lediglich die Nutzungsrechte daran lassen sich erwerben. Modifikationen sind nicht gestattet, selbst dann nicht, wenn sich das Programm auf Ihrem Rechner erst gar nicht zum Laufen bringen lässt. Eine Funktionsgarantie gibt es weder für Ihre aktuelle und schon gar nicht für Ihre zukünftige Computerperipherie.
- Zuspiel 2: weiter, pause
- Sprecher 2: Wenn Sie auf Ihrem Computer ein Microsoft Office-Paket älteren Datums installiert haben, verlieren Sie mit der Installation eines aktuellen Word, das ja auch im Office-Paket enthalten ist, die Nutzungsrechte an Ihrem alten Office, da das Paket nicht mehr mit allen Komponenten in ihrer ursprünglichen Version auf Ihrem Rechner vorhanden ist. Die Benutzung von Excel oder Outlook wird illegal.
- Zuspiel 2: weiter, pause

Sprecher 1: Unser Rechtssystem ist mittlerweile so kompliziert, dass ein Leben ohne Regelverstöße nahezu unmöglich scheint. Kann es aber im Sinne einer Demokratie sein, einen Großteil seiner Mitglieder, - auch und nicht zu selten - Kinder zu kriminalisieren?

Zuspiel 2: weiter

Sprecher 2: [über Zuspiel] Grundgesetz der Bundesrepublik Deutschland Artikel 14, Absatz 2: Eigentum verpflichtet. Sein Gebrauch soll zugleich dem Wohle der Allgemeinheit dienen.

Zuspiel 2: stop

Zuspiel 3: O-Ton Ed Moglen: *Because the recognition of individual possibility, to allow each to be what she and he can be rests inherently upon the ability of knowledge. The perpetuation of ignorance is the beginning of slavery.*

Sprecher 1: Die digitale Kopie und die Bandbreite von Internet und drahtloser Kommunikation ermöglichen eine weltweite Verfügbarkeit von Wissen und Kultur ohne zusätzliche Kosten, so die Argumentation der Free Software-Bewegung. Dabei distanziert sich die Initiative von jeglichem Vorwurf des Utopismus, sondern entwickelt ganz im Gegenteil konkrete Modelle für einen angemessenen pekuniären Ausgleich zwischen Erhebern und Nutzern, deren Ausführung hier zu weit führen würde. Genaugenommen muss man auch unterscheiden zwischen der politisch-philosophisch motivierten Free Software-Bewegung, einer deren Vorsprecher der eben gehörte Rechtsprofessor Ed Moglen ist, und der eher pragmatisch argumentierenden Open-Source-Initiative. Dass Software grundsätzlich kostenlos sein sollte, steht dabei für keine der beiden Varianten im Vordergrund. Entscheidend ist vielmehr die Offenlegung von Programmcode zu Lern-, Anpassungs- und Weiterentwicklungszwecken.

Sprecher 2: Die Geschichte Freier Software beginnt mit der Gründung des GNU-Projekts 1984 am Massachusetts Institute of Technology durch Richard Stallman. Stallman ärgerte sich, dass ein Abteilungsdrucker über Papierstaus oder fehlendes Papier keine Meldung lieferte. Er nahm sich vor, die Druckersoftware umzuschreiben und das fehlende Feature einzubauen. Bei der Herstellerfirma Xerox fand Stallman jemanden, der den Quelltext hatte, ihn jedoch nicht herausgeben durfte, da er eine Nichtweitergabe-Verpflichtung unterschrieben und sich damit zur Nichtkooperation verpflichtet hatte. Die Software war proprietär. Schlagartig wurde klar: Die paradiesischen Zustände der Freiheit und offenen Kooperation der Anfangszeit der Computerei waren endgültig vorbei. Damals war es selbstverständlich, dass der Quelltext von Computerprogrammen frei ausgetauscht und diskutiert wurde. Es war so üblich wie eine Veröffentlichung einer Forschungsarbeit in einer anderen wissenschaftlichen Disziplin. (LINUX & CO. Freie Software – Ideen für eine andere Gesellschaft. Stefan Meretz, 2000)

- Sprecher 1: Dass offene Software bis vor 20 Jahren selbstverständlich war, ist trotz der gegenwärtigen, erhitzten Debatten zu den Themen Freie Software und Software-Patent noch nicht ins allgemeine Bewusstsein gerückt. Heute steht Freie Software für eine moderne Variante des Aufklärungsgedankens. Wissen muss allgemein verfügbar sein, damit andere darauf aufbauen können. Wer sich abschottet, wird sich langfristig nicht weiterentwickeln können. Ein Gedanke, dessen Richtigkeit selbst die Industrie langsam begreift. Auch die Dienstleistungsgewerkschaft Verdi macht mobil auf Ihrer sehr empfehlenswerten Website:
- Sprecher 2: Die Benutzung proprietärer unfreier Software zwingt andere zur Benutzung derselben Software.
Die Benutzung proprietärer unfreier Software fördert die Monopolbildung
Die Benutzung proprietärer unfreier Software fördert Computertanphabetismus.
Die Benutzung proprietärer unfreier Software gibt dem Hersteller soziale Macht über Sie und andere.
- Sprecher 1: Wenn es um demokratische Grundprinzipien wie Freiheit und Gerechtigkeit geht, ist die Kunst nicht fern. Und so setzen sich zahlreiche Projekte, vorwiegend aus den Bereichen Medienkunst, Literatur und Bildender Kunst bereits mit dem Thema auseinander. Eine repräsentative Zusammenstellung finden Sie beispielsweise in der Public Library auf der Website der „Wizards of Os“, Os mit „s“ für Open Source. Wie aber steht es mit der abstraktesten aller Kunstformen, der Musik, die sich gerade im zeitgenössischen Bereich oftmals des Vorwurfs anachronistischer Tendenzen erwehren muss?
- Zuspiel 4: O-Ton Martin Rumori: *Auch was das Lernen angeht, ist Open Source einfach ganz wichtig, weil man einfach, wenn ich was nicht weiß gucke ich irgendwo rein, wo ich weiß, da wird's sicher so ähnlich gemacht, und dann kann ich davon lernen. Und da, glaube ich, tappt man sonst ziemlich im Dunkeln und findet vielleicht auch keine richtig gute Lösung, oder fragt sich immer, Moment, macht man das jetzt so, oder macht man es nicht so.*

- Sprecher 2: Klangliche Grundlage der interaktiven Klanginstallation *Grenzenlose Freiheit* bilden Aufnahmen aus städtischen Kontexten. Jeder dieser Kontexte wird durch drei auditive Ausschnitte repräsentiert. Gesteuert wird die Installation durch drei drahtlose Handheld-Computer, sogenannte PDAs, die die Besucher am Eingang erhalten. Jedem Computer ist einer der städtischen Kontexte zugeordnet. Der Besucher hat die Möglichkeit, über den Touchscreen des PDA einen der drei Klänge seiner Stadt auszuwählen und seine Eigenschaften zu verändern. Außerdem kann das Klangmaterial frei im Raum positioniert werden. Dazu werden 24 Lautsprecher verwendet, 8 in jedem der drei aneinanderhängenden Räume. Als Orientierungshilfe zeigt der PDA sowohl den Grundriss der drei Räume als auch die Standorte der Lautsprecher. Die momentane Position sowohl des eigenen Klangs als auch der anderen Klänge kann so auf dem Display nachvollzogen werden.
- Sprecher 1: Hören Sie bitte zunächst einen dieser städtischen Kontexte: eine Fahrt mit der Berliner U-Bahn im Original.
- Zuspiel 5: Grenzenlose Freiheit: Original (Privataufnahme der Komponisten)
- Sprecher 1: Und nun die durch einen Besucher manipulierte Version:
- Zuspiel 6: Grenzenlose Freiheit: Demo (Privataufnahme der Komponisten)
- Sprecher 1: Martin Rumori zu den programmiertechnischen Hintergründen seines zusammen mit Daniel Teige bei den diesjährigen „Wittener Tagen für Neuer Kammermusik“ realisierten Projektes:
- Zuspiel 7: O-Ton Martin Rumori: *Durch diese Idee, da gibt's einfach nichts, womit man das einfach so machen kann, die Software auf den PDAs ist zum Beispiel eben von Null geschrieben, und da war's halt durch meinen Hintergrund einfach nötig, ein System zu haben, auf dem ich arbeiten kann. Und das ist eben Linux, auf dem ich auch entwickeln kann. Also haben wir uns den Linux-PDA, den ersten Linux-PDA, Sharp-Zaurus war das vor zwei Jahren, gekrallt und mit dem gearbeitet. Ja, das ist so ein Beispiel, wie man jetzt mit geschlossenen Projekten, es sehr sehr schwer gehabt hätte, einfach aus finanziellen Gründen, wenn man sich jetzt irgendwie so ein Windows CE-Gerät gekauft hätte, man brauch dann die Entwicklungsumgebung, fängt an, neu zu lernen, wahrscheinlich dann nur für Windows CE, kann das nicht übertragen auf das Desktop-Windows, nehme ich an, weiß ich nicht, und da ist es so, ich entwickle auf meinen Laptop ganz normal die Software, lass die auf meinem Laptop laufen und am Ende lasse ich's noch mal bauen für die andere Architektur, für den Sharp Zaurus und das läuft da ganz genauso.*
- Sprecher 1 Rumori kommt hier auf ein Kriterium zu sprechen, das viele Komponisten als ein entscheidendes Argument für offene System anführen: Die Übertragbarkeit von Software auf verschiedenste Geräte unterschiedlicher Generationen, programmiertechnisches Know-How vorausgesetzt. Eine nicht unwesentliche Bedingung für die Beständigkeit eines medialen Kunstwerks. Der Holländer Michel Koenders:

- Zuspiel 8: O-Ton Michel Koenders: *Der Atari war auf einmal nicht mehr da. Ich habe dann eigentlich Wochen spendiert, um eigentlich wieder die Software umzuschreiben. Und das finde ich eigentlich grausam. Es gibt immer noch alte Kompositionen, die laufen eigentlich immer nur noch auf dem Atari, also ich habe noch ein paar zu Hause, und das macht eigentlich keinen Spaß. Also ich habe immer doch an sich Standard benützt, ob C oder Lisp, aber es gibt immer wieder Probleme, das wieder nach den neuen Maschinen zu machen.*
- Sprecher 1: Für Koenders ist es selbstverständlich, seine Programme öffentlich zugänglich zu machen. Zur Zeit arbeitet er am Zentrum für Kunst und Medientechnologie, kurz ZKM in Karlsruhe an einem abendfüllenden Werk ...
- Zuspiel 9: O-Ton Michel Koenders: *„Cyclotron“ heißt das eigentlich, und es geht über das Zusammenprallen, eigentlich auf verschiedenen Ebenen: mikro, meso, makro, also die ganz große Sache, die ganz kleine Sache eigentlich, aber auch auf unserer Ebene, also mehr oder weniger sozialen Ebene.*
- Sprecher 1: Bouncing Balls I ist eine Studie daraus.
- Sprecher 2: Die Grundlage dieser 4-teiligen Komposition bilden leicht in Bewegung versetzte Metallkugeln auf den Saiten im Klavier. Diese Kugeln verhalten sich wie Mobiles, die sich in permanenter Bewegung auf der Suche nach einer Stabilität befinden, die sie niemals erreichen werden. Diese konstante Bewegung zieht sich durch das ganze Stück, kontrapunktiert von einer einzelnen unveränderten Note.
- Zuspiel 10: Michel Koenders: Bouncing Balls (Klavier – Nora Mulder. Privataufnahme des Komponisten)
- Zuspiel 11: O-Ton Torsten Anders: *Ich arbeite eigentlich fast ausschließlich mit offener Software, auch für alle Sachen, wie Texte schreiben und so weiter. Und das hat hauptsächlich zwei Gründe: zum einen ist man mit offener Software sehr unabhängig. Ich habe mit verschiedener Software auch von verschiedenen Instituten gearbeitet, und da ist es so, also zum Beispiel Software vom IRCAM, einem sehr wichtigen Institut aus Frankreich: entweder man kauft jedes Jahr neu die Lizenz dafür oder man hat die Software dann nicht mehr. Und wenn man mehrere Monate oder gar Jahre in die Entwicklung von eigenen Programmen gesteckt hat, die auf andere Programme aufbauen, will man die sich nicht wegnehmen lassen, nur weil man die Lizenz nicht nachgekauft hat. So ähnlich ist es, wenn man updated, und dann plötzlich die Software nicht mehr läuft, was ja durchaus möglich ist*
- Sprecher 1: ... der zur Zeit am Sonic Arts Research Center der Queen's University in Belfast an seiner Doktorarbeit sitzende Komponist Torsten Anders. Bereits seit einigen Jahren setzt auch das schon erwähnte ZKM in Karlsruhe auf die Flexibilität von Linux und offener Software. Ludger Brümmer, Komponist und Leiter des dortigen Instituts für Musik und Akustik:
- Zuspiel 12: O-Ton Ludger Brümmer: *Sicherlich motiviert zum einen, dass die Systeme nicht mehr so speziell sind, zum anderen auch durch die Public Domain, durch diese große Free Software-Bewegung, die einem eine Verfügbarkeit einerseits ermöglicht, aber auch andererseits eine Öffnung und eine Verbindung verschiedener Elemente. Also wenn man zum Beispiel sieht: eine Software benutzt zur Kommunikation eine Sprache, die auch eine andere Software benutzt. Also diese Softwares heutzutage sind nicht mehr exklusive abgekapselte Systeme, sondern sie interagieren mit vielen anderen Softwares auch, oder sie haben Möglichkeiten, diese Interaktion zu implementieren.*
- Sprecher 1: Dieser Flexibilität steht in den meisten Fällen eine äußerst dürftige Dokumentation der Programme gegenüber. Ein Blick in den Quellcode wird unausweichlich. Oder – und hier liegt ein entscheidender Pluspunkt der Open-Source-Community, man wendet sich an eine der in der Regel äußerst hilfsbereiten und geduldigen Mailing-Listen. Jede Email wird hier an jeden Teilnehmer geschickt: ein ebenso einfaches wie effektives Diskussionsforum.
- Zuspiel 13: O-Ton Torsten Anders: *Gerade in diesen Tagen ist mir das so gegangen, dass ich mit einer Sound-Synthese-Sprache gemerkt habe, Feature xy unterstützt das nicht, und habe mich mit einer Frage an die entsprechende Mailingliste gewandt. Und siehe da*

eine Stunde später kriegte ich ein Update von einem C++-File, was dann zu diesem Programm gehört, wo mir gesagt wurde: versuch' das mal, ob das jetzt schon geht. Das erlebt man bei keinem kommerziellen Hersteller, dass man so einfach, so schnell das Programm selber, das man benutzt, seinen eigenen Zwecken anpassen kann.

- Zuspiel 14: O-Ton Ludger Brümmer: *Es geht nicht nur um Hilfe, sondern auch, es geht darum, dass jeder, der die Software benutzt, daran eine Verantwortung übernimmt, um Fehler zu finden, Fehler zu identifizieren, und dadurch ein Feedback an den Programmierer gibt, ihm auch Zeitsparen hilft, indem der ihm den Fehler herausucht, soweit es ihm möglich ist. Und dann gibt es meistens eine schnelle Reaktion darauf und dann gibt's eine neue Version und die hat dann den Fehler korrigiert. Es gibt auch Momente, wo der Programmierer selber nach neuen Features fragt, was wollt ihr, was hättet ihr gerne noch implementiert, und das finde ich eine unglaublich usernahe Idee. Gerade der Open Community wird ja oft ihre User-Feindlichkeit vorgeworfen, weil die Dokumentation so schwierig ist, und weil teilweise sehr kryptische Bedienungsformen angedacht sind in der Software. Wenn man diesen Aspekt gegenüberstellt, dann finde ich, lässt sich dieser Vorwurf nur begrenzt aufrecht erhalten. Es gibt einen sehr engen Kontakt zwischen den Usern und den Entwicklern. Und das Besondere ist einfach, das die Denkweisen andere sind. Das finde ich das Interessante. Es wird nicht gedacht daran, ich habe ein Produkt und ich habe jetzt einen Markt und muss jetzt für jeden in diesem Markt das Produkt optimieren und mit anderen Worten: das Mittelmaß finden und implementieren. Sondern ich kann einfach weiterhin extrem denken und sagen: Ich will das machen, was es noch nicht gibt, oder was verrückt ist, oder einfach was klug ist. Also der kommerzielle Entwickler muss immer denken: will mein Kunde das, oder hat das eine Chance auf dem Markt. Alles was keine Chance hat fliegt raus. Und das- ich sag mal so ein bisschen - lässt die User verdummen, weil sie gar nicht weiterdenken dürfen.*
- Zuspiel 15: O-Ton Thomas Grill: *Mir ist es im Prinzip darum gegangen, eine Art von Abschluss mit meinem Schlagzeugerdasein zu machen, also eine Art von Wiederbeschäftigung mit einem bestimmten Instrument sein, eben dieses Beckens, das da solistisch eingesetzt wird, und featuret einerseits eine gewisse Spielart von Schlagzeugstil, die ich bevorzuge, also sehr arhythmisch, kann man sagen, ein Rhythmus, der sehr impulsiv ist und andererseits aber eine gewisse Klangkultur, die mir wichtig ist, hervorzeigt. Und, ja, also im Prinzip geht's auch, ja es ist eine Dehnung der Zeit in gewisser Weise. Also das Stück könnte länger dauern, tut es aber nicht, es bricht ab, bedeutet aber auch, dass es eine Fortsetzung geben könnte, in irgendeiner gewissen Weise auch mit dem Schlagzeugspiel in einer anderen Form.*
- Zuspiel 16: Thomas Grill : Ride (CD glear, Karate Joe Records KJ013)

Sprecher 1: Das war *Ride*, eine Komposition des Wieners Thomas Grill, der vom Programmieren lebt. Er hat wesentliche Beiträge für die Audio-Entwicklungsumgebung Pure-Data geleistet und differenziert sehr genau zwischen Dingen, die er frei zur Verfügung stellen möchte und solchen, die er lieber für sich behält:

Zuspiel 17: O-Ton Thomas Grill: *Ich glaube, dass für die meisten Menschen es so ist, dass die Zeit des Profitierens eine ziemlich lange Phase ist. Also einmal nur schauen, zuhören, möglicherweise auf Listen lauschen, Mailing-Listen, was es da so gibt an Information. Ich glaube, es ist auch sehr wichtig, einmal zu sehen, wie andere Leute arbeiten in der Open-Source-Szenerie, wie Austausch funktioniert, unter Umständen auch soziale Komponenten, die damit verbunden sind. Und irgendwann kommt dann der Punkt, wo man vielleicht selbst da produktiv wird. Und es ist natürlich nur logisch, auch der Community dann etwas zurückzugeben von dem Wissen, das man angesammelt hat durch Recherche und so weiter ... Und das ist für mich schon ein wichtiger Punkt. Also ich würde mich nicht als Open-Source-Fanatiker bezeichnen, oder als politischen Open-Source-Menschen. Es ist ein mehr pragmatischer Zugang, also ich habe etwas gelernt und ich gebe auch der Community wieder etwas zurück, wobei es durchaus auch sein könnte, dass das Zurückgeben auch nur ein Teil dessen ist, was ich mache. Also ich denke mir, gewisse Dinge sind für die Open-Source-Community wichtig oder brauchbar, die will ich zurückgeben. Und Dinge, die aber sehr persönlich sind zum Beispiel für die Realisierung von einzelnen Stücken würde ich nicht unbedingt öffentlich verfügbar machen.*

Sprecher 1: Szenenwechsel. Noch vor wenigen Jahren war der Alltag elektroakustischer Musikproduktion - und zwar unabhängig von U oder E - unweigerlich gekoppelt an Studios mit millionenschwerem Equipment. Für dessen Bedienung standen Toningenieure zur Verfügung, die sich mit ihren „Instrumenten“ bestens auskannten und ihren persönlichen Stil im Umgang damit entwickelten. Ob der sorgsam vergebenen, wertvollen Studiozeit und der Schwierig- wohl auch Schwerfälligkeit der Benutzung, musste ein Komponist, sich auf eine parasitäre Beziehung zu seinem Ingenieur einlassen, ob er es wollte oder nicht. Es liegt nahe, dass dies zuweilen zu Konflikten bezüglich der Urheberschaft von Ideen führte. So gibt es zur Entstehung von Stockhausens „Mantra“ mindestens zwei Versionen. Die des Komponisten und die des Studioleiters.

Zuspiel 18: O-Ton Ludger Brümmer: *Die junge Komponistengeneration lernt inzwischen selber mit Software umzugehen. Das ist nichts besonderes mehr, das ist selbstverständlich geworden. Deswegen erwarte ich eigentlich auch von den Komponisten, dass sie selbst den Computer bedienen können. Wir würden zum Beispiel für bestimmte Fragestellungen oder Probleme jemanden zur Seite stellen, um das Problem zu lösen, aber die Begleitung durch die Komposition oder die Entwicklung der Komposition führt der Komponist selber durch. Und aus eigener Erfahrung weiß ich, dass dieser Dialog mit dem Computer und der Software, also das Erkennen von Möglichkeiten und das Empfinden von Potentialen, unglaublich wichtig ist für den Kompositionsprozess. Also dass ich darüber nachdenke, wie ich bestimmte Probleme lösen kann. Und zwar in der Technik. Es gibt ein Problem in dem Moment, wo ich nicht weiß, womit ich umgehe. Also in dem Moment, wo ich einen Vermittler habe, der zwischen mir und dem Instrument steht, wird es ein Stück schwieriger zu begreifen, welche Potentiale in dem Instrument stecken. Und das ist, glaube ich, das, was die Fantasie so sehr anregt, dass ich träume, dass ich mir im Halbschlaf oder in der Müdigkeit einfach irgendeinen Code irgendwelche Bedienelemente vorstelle oder Möglichkeiten entwickle. Und dieses Träumen von den Instrumenten und deren Möglichkeiten, das ist eine ganz wichtige Sache, und das funktioniert nur, wenn ich selber damit direkt umgehe. Also das ist auch ein bisschen, selber in die Hand nehmen wollen. Das ist, glaube ich, auch eine Sache. Das An-einen-Vermittler-Übergeben, hat auch immer zu tun mit Abgeben und auch Distanz. Und ich glaube auch, das spürt man in der Sprache, die diese Kompositionen, die so entstanden sind, entwickelt haben. Denn da hat man eher das Gefühl, diese Sprache bleibt bei dem in Anführungsstrichen konventionellen instrumentalem Denken, sie geht gar nicht so sehr auf die Möglichkeiten des Instrumentes ein. Oder man könnte jetzt noch weiter gehen und sagen: der Vermittler ist ja eine eigene Instanz. Und der Vermittler drückt natürlich auch der Komposition seine Vorstellungen auf. Und ich bin mir nicht sicher, ob das immer zum Vorteil für die Komposition ist, und ob der Komponist da das kriegt, was er eigentlich will.*

Sprecher 2 : In Ludger Brümmers *temps du miroir* wird der Versuch unternommen eine dichte Struktur aus vorbereiteten und live interaktiv erzeugten Klängen mit dem Klavier als Mittelpunkt zu erzeugen. Der Spiegel ist hier die Metapher für die Spiegelung des Instrumentes Klavier auf vielfältiger Art und Weise. Weiter gehend arbeitet diese Spiegelung nicht mit der zeitgleichen Reflektion des Klavierklanges, sondern greift in die Darstellung (die Spiegelung) aktiv ein, um sowohl die zeitliche Parallelität, die spiegelverkehrte exakte Reproduktion, als auch die klangliche Parallelität aufzuheben. Die Verwendung der Spiegelung in *temps du miroir* verzerrt, ändert die Perspektive aber vor allem löst sie das Spiegelbild aus seiner zeitlichen Abhängigkeit zum Original, lässt es selbstständig werden und wird selbst zum Original, welches vom ursprünglichen Original gespiegelt wird.

Zuspiel 19: Ludger Brümmer: temps de miroir (Maki Mamekawa - Klavier, Joachim Goßmann – Programmierung und Live Elektronik, Ludger Brümmer - Live Elektronik. Mitschnitt der Aufführung vom BR, ZKM 03.06.2004)

- Sprecher 1: Wie der in den USA wirkende gebürtige Kroatie Ivica Ico Bukvic eingangs bemerkte, stimuliert jede Software zu einem ihr eigenen intuitiven Gebrauch. Somit ist das Zugreifen auf den Toningenieur mit seinem Equipment in künstlerischer Hinsicht vorsichtig mit der Verwendung von „fertiger“ Software vergleichbar. Ihr Potential für einen individuellen Einsatz ist begrenzt. Das ist im Prinzip nichts Neues: Auch die traditionelle Instrumentalgeschichte kennt ähnliche Probleme. So produzierte der Klavierbau des 19. Jahrhundert unzählige, heute mehr oder weniger vergessene Varianten dieses Instrumentes, bis sich zum Jahrhundertwechsel ein schlichter, bis heute gültiger Standard etablierte. Nach dieser Entwicklungsstagnation beschäftigte sich das 20. Jahrhundert in Form mannigfaltiger Spiel- und Präparationstechniken mit dem konnotationsbehafteten Klangerzeuger. Bei all dem historischen Ballast ist ein neues, frisches Klavierstück heute nur noch schwer vorstellbar. Allerdings ist die Halbwertszeit von Software, ja die Halbwertszeit elektronischer Musik um ein Vielfaches kleiner als die akustischer Instrumente. Und so verblüfft es wenig, dass bereits nach wenigen Jahren viele Programme und Methoden ihr kreatives Potential verbraucht haben, dass viele ältere Kompositionen abgenutzt und klischeehaft erscheinen.
- Sprecher 2: Für das mehrkanaligen Tonbandstück „With Shifting Joints“ hat Torsten Anders instrumentale Gesten und physikalische Artikulationen komponiert, die auf physikalische Modelle, sozusagen virtuelle Instrumente angewandt wurden. Solche Instrumente erlauben dem Komponisten die Realisation unkonventioneller Artikulationen auf einem hohen Genauigkeitsniveau und extreme Parameterveränderungen. Einige der Artikulationen im Stück sind physikalisch unmöglich auf einem akustischen Instrument aufzuführen. So wurden zum Beispiel zwei frei schwingende Saiten durch eine Sprungfeder verbunden, was ein zusammengesetztes Instrument mit einem komplexen Frequenzspektrum ergab. Während der Anregung einer der Saiten mit einem Bogen, wurde die Feder dann entlang der Länge der Saiten verschoben, was eine drastische Wirkung auf die Obertonentwicklung des Instrumentenklangs zur Folge hatte.
- Sprecher 1: Leider ist die 8-kanalige Version von *With shifting joints* im Radio nicht reproduzierbar. Auch hätte eine Heruntermischung auf zwei Kanäle stark verfremdende Nebeneffekte zur Folge. Eine reduzierte Fassung mit zweien der acht Originalkanäle vermittelt da einen besseren Eindruck:
- Zuspiel 20: Torsten Anders: With shifting joints (Privataufnahme des Komponisten)
- Zuspiel 21: O-Ton Torsten Anders: *Ich bin das erste Mal über Tao gestolpert zufällig in einer Mailingliste 1999, glaube ich, oder 2000. Ich habe das Programm studiert, ich kannte zu dem Zeitpunkt schon andere Programme, die physikalische Modelle unterstützen, Tao war in vielen Aspekten anders und ich hab dann sehr viele verschiedene Versuche damit gemacht, was man so damit machen kann, habe ich ausprobiert. Und dann ist im Lauf von Jahren, muss ich schon sagen, hab ich rundherum um Tao dann mir meine eigene Kompositionsumgebung gebaut. Das heißt, Tao ist skriptgesteuert, ich kann Partituren für Tao erzeugen, die dann das Programm ausführt, und diese Partituren, diese Skripte habe ich wiederum erzeugt mit eigenen Programmen.*
- Sprecher 1: Und noch ein Szenenwechsel. Der – so ist zumindest zu vermuten – ursprüngliche Zweck von Notation war, auf anderem Wege tradierte, gespielte oder gesungene Musik aufzuzeichnen, um sie landläufig zu verbreiten und kommenden Generationen verfügbar zu machen. Nachdem aber einmal ein Notationssystem erfunden war, lag es nahe, mit ihm Musik zu erfinden, die dann gespielt werden konnte. Was ich damit sagen möchte: Im Idealfall beginnt der Produktionsprozess einer Komposition mit einer Idee und einem leeren Blatt - im Gegensatz zu

dem althergebrachten Entstehungsprozess elektronischer Musik, der sich von Studio- und Softwareperipherie inspirieren lässt. Ich gebe es zu, dieser Vergleich hinkt ein bisschen. Auch ist durchaus diskutierenswert, was denn eigentlich der Idealfall der Entstehung eines Kunstwerks ist. Worauf ich aber hinaus will: Ein Komponist, der selbst programmiert, macht sich unabhängig von den Beschränkungen vorhandener Software, und die Idee, das leere Blatt, oder in diesem Fall: der leere Bildschirm, kann wieder am Anfang der Entwicklungskette zu stehen.

Zuspiel 22: O-Ton Michel Koenders: *Ja, eigentlich war das Idee schon da und eigentlich auch wie es aussehen sollte und wie es funktionieren sollte. Da habe ich Zeichnungen gemacht, ich hab's aufgeschrieben. Und eigentlich seit erst drei oder vier Monaten beschäftige ich mich mit Programmieren, das wirklich alles hinzubekommen. Die Noten gehören auch dazu. Ich hab schon vor vielen Jahren eigentlich ein Programm in Lisp geschrieben, um algorithmische Komposition zu machen ...*

Zuspiel 23: O-Ton Ludger Brümmer: *Normalerweise wechsele ich meine Programme nicht ständig, also weil ich so ein Programm erstens danach aussuche, wie vielfältig es ist und wie allgemein es ist. Also deswegen ist mir eine Programmiersprache, die ich selber programmieren kann am liebsten, weil ich hab da nur eine Anzahl von Elementen, die ich selber konfigurieren kann. Also eine relativ große Freiheit. Da entstehen zwar Nachteile in der Bedienung, also das Programmieren muss natürlich gemacht werden und dauert seine Zeit, aber dann habe ich für eine sehr lange Zeit etwas sehr Allgemeines. Genauso wie ich die Sprache Deutsch habe, die aus bestimmten Wörtern besteht, ein paar tausend Wörter, und mit der ich mein Leben lang alles ausdrücken muss, neben der gestischen Artikulation, was ich denke und was ich fühle. Und so ähnlich sehe ich dann bestimmte Elemente, bestimmte Software, die ich benutze. Wenn ich also so eine Sprache benutze, dann brauche ich schon am Anfang eine Idee. Dieses Instrument einer Sprache ist so allgemein, dass ich ohne die Idee, etwas ausdrücken zu wollen, nichts sagen kann. Also muss ich vorher einen Gedanken haben, mit dem ich an den Computer herantrete, und den ich dann umsetze.*

- Sprecher 1: War der Zugang zu Open Source bei den bisher vorgestellten Komponisten überwiegend pragmatischer Natur, so wird er bei dem an der Freiburger Musikhochschule lehrenden Orm Finnendahl zum kompositorischen Konzept:
- Sprecher 2: Bei *grep* handelt es sich nicht um eine Komposition im traditionellen Verständnis einer Fixierung von Spielhandlungen und den damit verbundenen Klängen der Musiker. Fixiert wurden vielmehr elektronische Verfahrensweisen und syntaktische Zusammenhänge, die festlegen, wie der Computer von den Interpreten gespielte Klänge verarbeitet. Die Klänge selber werden weitgehend von den Musikern selbst bestimmt. Das bedingt einen veränderten Proben- und Entwicklungsprozess des Stückes, in dem die instrumentalen Spielweisen und das Programm in wechselseitigem Bezug entwickelt werden. Damit verbunden ist die Anforderung an die Interpreten, ihrerseits aktiv an der Gestaltung der Aufführung mitzuwirken.
- Zuspiel 24: Orm Finnendahl: *grep* (Sascha Armbruster - Saxophon, Burkhard Beins – Schlagzeug, Orm Finnendahl – Programmierung, Aufnahme SWR)
- Zuspiel 25: O-Ton Orm Finnendahl: *Also ich möchte versuchen zu erklären, wie bei mir der Open Source-Gedanke sich dann umgesetzt hat. Ich hab' bei dem Stück, das ich jetzt hier für das ZKM realisiert habe, die Idee gehabt, dass ich bei der Komposition nicht eine fertige Komposition mitbringe, die die Musiker oder Interpreten dann lediglich umsetzen müssen, sondern dass ich von meiner Seite aus eine Art Framework, heißt das im Englischen, also eine Art von Rahmen, in dem die Musiker sich bewegen können, da gibt es also dann formale Festlegungen, wann bestimmte Dinge, die die Musiker spielen wiederholt werden von der Elektronik, also sie werden dann mitgeschnitten und dann wiederholt, wann Dinge in welcher Form wie transformiert wiederholt werden, und diese ganzen Dinge habe ich mitgebracht, es war aber zunächst vollkommen unklar oder in gewisser Weise nicht festgelegt, was die Musiker im Konkreten spielen. Und dieses Material, mit dem gespielt wurde, musste dann erst in der Probenarbeit gefunden werden. Gleichzeitig hab' ich das ganze Framework, also diesen Rahmen so flexibel gestaltet, dass ich ihn auch permanent verändern konnte. Und dadurch konnte sich in so einer Art Annäherungsprozess irgendwie das Stück entwickeln. Und ich würde sagen, das ist vielleicht nur eine Metapher, aber für mich ist es so, dass es so etwas Ähnliches wie Open Source ist. Ich hab' sozusagen meine Quellen mitgebracht. Ich hab' jetzt nicht ein fertiges Stück mitgebracht, nicht gesagt, guck' mal, was hier für ein tolles Stück ich jetzt schon habe in meinem Kämmerchen ausgedacht, sondern ich habe eine Art Framework, einen Rahmen mitgebracht, hab' meine Quellen, wie ich kompositorisch arbeite, praktisch offen gelegt und habe dann die Musiker quasi eingeladen oder ihnen offen gelassen zu sagen, passt auf, geht damit um. Ihr müsst irgendwie meine Grenzen, die ich habe oder die ich setze, die müsst ihr akzeptieren, aber ihr habt eure eigenen Möglichkeiten und hab' das versucht so hinzubekommen, dass die Musiker ihre Qualitäten, die sie ja zweifellos haben, wo sie viel mehr vielleicht auch Verständnis von haben als ich, nämlich von ihrem eigenen Instrument, von dem wie Klänge produziert werden, dass sie das, diese positiven Dinge einbringen können und ich bringe das ein, was ich gut kann, und man in so einer Art von gemeinsamer Arbeit dann vielleicht zu einem Ergebnis kommt, was dann auch nach außen hin deutlich wird, dass es funktioniert.*
- Zuspiel 2: Ivica Ico Bukvic: Legisonitus I "Gone in 8 minutes and 11.527 seconds"

Sprecher 1: [über Zuspil] Die zunehmende Verbreitung von Linux und die aktuelle tagespolitische Berichterstattung über Freie Software spiegeln in den Augen ihrer Protagonisten den Beginn einer langfristigen gesamtgesellschaftlichen Entwicklung wider, die dem Einzelnen größere Selbstbestimmung, mehr Demokratie und Unabhängigkeit garantiert. Wie so oft in der Geschichte geht es hier um Freiheit und Entfaltung, beides wesentliche Bedingungen für den Fortbestand unserer Kultur. Der marktwirtschaftlich in Slogans wie „You make it a Sony“ längst vereinnahmte Begriff von Individualität bekommt wieder eine Perspektive. Das stark angekratze „cogito ergo sum“ der Aufklärung, die Verantwortlichkeit des Einzelnen als Selbstverständnis menschlicher Existenz erfährt eine Renaissance.

Die um ein Vielfaches gestiegene Teilnehmerzahl der in diesem Jahr zum zweiten Mal im ZKM in Karlsruhe abgehaltenen Linux-Audio-Konferenz zeigt auch im musikalischen Bereich ein wachsendes Interesse an einer kommunikativen Zusammenarbeit, ein Verlangen nach unverfälschter Individualität, frei von dem Zwang zu Produkten und Features, von denen andere meinen, man müsse sie brauchen. Auch wenn nicht jede im Bereich der Open-Source-Welt entwickelte Software einem Vergleich mit ihrem kommerziellen Pendant bereits standhält, auch wenn nicht jede in einer Komposition zum Einsatz kommenden Software den Anspruch individueller Klanggestaltung und Originalität erfüllt, den man gerne an sie stellte.

Zuspil 2: weiter

Sprecher 1: [über Zuspil] Ivica Ico Bukvics *Legisonitus I „Gone in 8 minutes and 11.527 seconds“* ist nicht unbedingt ein repräsentatives Stück für den Komponisten, der zur Zeit am Oberlin College sein PhD ablegt und eine sehr aktive Mitglied der Linux-Audio-Developer ist. Repräsentativ ist es in jedem Fall als Beispiel der intuitiven Benutzung einer Hard-beziehungsweise Software.

Zuspil 2: stop

Zuspil 26: O-Ton Ivica Ico Bukvics: *And this particular piece is this very extreme concrete piece, where I just took my microphones, hooked them up to my glasses and want to pick up my wife from work and just recorded the whole thing*