

Golo Föllmer

networkshop(ping) - Musikmachen im Netzwerk

Im Sommer 2000 sprach mich Frank Geißler an, ob ich für die Dresdner Tage der zeitgenössischen Musik ein Konzert mit Netzwerkmusik veranstalten wolle. Ich beriet mich mit Musikern und kam zu dem Schluss, dass eine klassische Bühnendarbietung, ein Konzert als Darbietungsform ungeeignet war. Stattdessen entschieden wir uns für einen zweitägigen Workshop: Ein Tag war für Schüler vorbehalten, ein zweiter Tag sollte Gelegenheit bieten, Konzeption und Praxis der Musiken einem interessierten Fachpublikum vorzustellen, wobei die Workshopteilnehmer abschliessend bei einer Performance mitwirken sollten.

Der ‚networkshop‘ wurde so umfangreich und aufwendig gestaltet, um den Besuchern einen möglichst breiten Einblick in das neuartige Phänomen ‚Netzmusik‘ zu bieten. Dies war aber nur der didaktische Teil der Überlegungen. Es gab darüber hinaus einen künstlerischen Grund, die Veranstaltung als Workshop zu konzipieren: Musik, die im Netz entsteht, muss sich auf die Bedingungen des Mediums einlassen. Die Gründe, warum mir ein Workshop beim momentanen Stand der Netzmusik als beste Darbietungsform für diese Musik erscheint, erschliessen sich aus den im folgenden dargelegten Bedingungen und künstlerischen Strategien, die Musiker daraus ableiten.

Die Bedingungen des Netzes lassen sich in zwei Typen unterscheiden: technische und kommunikative. Die technischen Spezifika legen in erster Linie fest, was möglich ist, während die kommunikativen Spezifika bestimmen, was für die künstlerische Auseinandersetzung interessant ist.

Zuerst zu den technischen Bedingungen.

Alle Computer-basierten Mittel, die für eine musikalische Arbeit im Netz zum Tragen kommen sollen, müssen technisch netzwerktauglich gemacht werden; bei einem Synthesizer z.B. muss die Kommunikation über ein Netzwerk möglich sein. D.h., die Information jedes Elements muss für die Übertragung über ein Netzwerkprotokoll erfasst und codiert werden. Im Fall des Consumer-Synthesizers muss nur der bestehende MIDI-Code in ein versandfähiges Internetprotokoll verpackt werden, andere Geräte und Softwaremodule erfordern die Entwicklung spezieller Datenumsetzer (zur Digitalisierung oder Filterung der relevanten Daten), eigener Codesysteme (zur Darstellung in der richtigen Form und Auflösung) und anwendungsgerechter Übertragungsprotokolle (zur

Übermittlung der Daten in der nötigen Art und Häufigkeit). Solche Entwicklungen schreiten seit Mitte der 90er Jahre voran, als das Internet begann, sich rapide zu verbreiten: Audio-Übertragung über das Netz wird mit wachsenden Bandbreiten immer echtzeitlicher und qualitativ besser. Audiovisuelle Formate (*Flash*, *Shockwave*, *Beatnik* u.a.) sind entstanden, die zunehmend komplexere Möglichkeiten der Audioverarbeitung und der Benutzerinteraktion bieten. Gängige Sequencer (*Cubase*, *Logic*), die auf dem musikalischen Kommunikationsstandard MIDI aufsetzen, sind mit Entwicklungen wie *Rocket Control* inzwischen für das (fast echtzeitliche) Zusammenspiel im Netzwerктаuglich. Avancierte Software aus dem Umfeld der akademischen Computermusik-Schmieden (*Max/MSP*, *Supercollider*) werden durch musikspezifische Übertragungsprotokolle wie *OSC* (Open Sound Control) für den Einsatz im Internet anwendbar. Computermusik-Konzepte werden durch Implementierung in der plattformübergreifenden Sprache *Java* (z.B. bei *Jsyn*) zu ‚mobilem Code‘, wodurch das große Problem der Kompatibilität überwunden wird, da *Java* auf nahezu allen Betriebssystemen und Rechnern ausgeführt werden kann. Keine der genannten Möglichkeiten der Netz-Audio-Technologie gab es im Jahr 1996, der Entwicklungsschritt der letzten fünf Jahre ist immens. Dennoch äußern sich viele Computermusiker unzufrieden mit dem heute technisch Machbaren. Sowohl Übertragungsbandbreite als auch Kompatibilität und Flexibilität der verfügbaren Mittel werden von vielen nach wie vor als unzureichend beurteilt. Die technische Entwicklung musikalischer Netzwerkzeuge steckt noch in ihren Anfängen.

Als zweites zu den kommunikativen Bedingungen, zum ‚Charakter‘ des Netzes.

Im Gegensatz zu den technischen sind die kommunikativen Bedingungen nicht objektiv gegeben, sondern bilden sich selbst durch Kommunikation heraus. Sie entspringen also einem rekursiven Regelprozess, der sich genauso an sozialen Bedürfnissen wie an technischen Vor- und Nachteilen des Mediums orientiert. So ist die Bühne des Netzes wie kein anderer Präsentationszusammenhang / kein anderes Medium öffentlich. Das Netz ist daher als Kommunikationsstruktur ideal zur teilnehmenden, aktiven Rezeption. Als Medium passiver Rezeption hingegen ist es u.a. wegen Einbußen bei der Audioqualität gegenüber qualitativ hoch entwickelten Audiomedien wie dem Radio relativ uninteressant. Auch die strikte Bindung an einen einsamen Stuhl vor dem Bildschirm schmälert den Erlebniswert der Netzwerkkommunikation. Hörer von Netzmusik sind also nicht nur technisch in der Lage, teilzunehmen, zu kommentieren und zu interagieren. Aktive Teilnahme ist quasi die ideale, ja fast die einzig sinnvolle Form der Rezeption im Netz.

Aktive Rezeption setzt aber Flexibilität beim rezipierten Objekt oder Vorgang voraus.

Ähnlich wie es Max Neuhaus 1974 für eine raumbezogene Musik außerhalb des Konzertsaals

konstatierte¹, muss eine Musik des öffentlichen Datenraums in einem solchen Maße offen sein, dass sie dem Rezipienten den Aktionsspielraum gibt, der seinem Umgang mit dem Raum/Medium entspricht bzw. auf diesen eingeht.

Maschinenbasierte Kommunikation und Rezeption ist einerseits durch sensorielle Beschränkung geprägt, ist für die Sinne eng und eindimensional. Gegen diese Ödnis arbeiten Computer- und Netzkunstwerke, die durch Grafikgewitter den Sehsinn irritieren und durch Navigationsverwirrung der starren Hypertext-Logik Sinnlichkeit entgegen setzen². Andererseits konkurriert die strikte Regelmäßigkeit der maschinellen Abläufe mit der eigenen zeitlichen und inhaltlichen Unvorhersehbarkeit der Maschine: mit dem Eigenleben und den Launen der Computer. Die Begegnung mit der Netzmaschine ist daher nicht linear narrativ, sondern prozesshaft und offen strukturiert.

In der Kunst ist Prozesshaftigkeit nichts Neues. Sie ist einerseits eine Methode, um überraschende Ergebnisse herzustellen. Andererseits wird damit eine Reproduzierbarkeit im Sinne Walter Benjamins vermieden und absolute Einmaligkeit hergestellt. Prozesshafte Kunst ist vorsätzlich flüchtig.

In Bezug auf Prozesshaftigkeit ist es nötig, zwischen zwei Ebenen des künstlerischen Codes zu unterscheiden. Auf der einen Seite gibt es den Phänocode, der die Struktur des wahrgenommenen Phänomens beschreibt. Dieser Phänocode ist in prozesshafter Kunst und Musik ephemere: variabel und flüchtig. Auf der anderen Seite steht der Genocode, der die Struktur des künstlerischen Erzeugungssystems darstellt. Dieser Genocode ist auch bei prozesshafter Kunst und Musik im Normalfall festgelegt, fixiert im Regelwerk des prozessierenden Systems.

Am Beispiel der nach Dresden eingeladenen Komponisten Guy van Belle, Chris Brown und Sergi Jordà möchte ich erläutern, wie Musiker auf die genannten Bedingungen eingehen: auf den Aspekt des Öffentlichen, die Distanziertheit der Bildschirmkommunikation, die Dualität von Regelmäßigkeit und Prozesshaftigkeit der Maschinenkommunikation und die daraus resultierende Vergänglichkeit der musikalischen Produkte.

¹ Vgl. Max Neuhaus: *Program Notes*, Toronto 1974; in: Max Neuhaus: *inscription – sound works volume I*, Ostfildern 1994, S.34.

² Schöne Beispiele sind <http://www.membank.org> und <http://www.jodi.org>

Sergi Jordà macht Audio-Software, die sowohl ein Instrument und Werkzeug als auch ein künstlerisches Statement darstellt. Sein Programm *FMOL (F@ust Music On Line)*³ bietet eine große Anzahl digitaler Syntheseformen, Effekte und Kombinationsmöglichkeiten. Es zeichnet sich durch ein schnell erlernbares Interface aus, das sehr flexibel ist und komplexe musikalische Improvisation erlaubt.

Ihre spezielle Prägung erhält die Musik durch ein weiteres Element, nämlich durch die Implementierung eines netzbasierten Apparates, der gestalterische Kollaboration auf besondere Weise befördert. Das Instrument erlaubt die Aufnahme kurzer Stücke (bei der ersten Programmversion maximal 20 Sekunden, bei der aktuellen Version eine Minute), deren digitale Partitur auf einen speziellen Server geladen wird und damit anderen *FMOL*-Spielern zur Verfügung steht. Wer sich eine der Partituren vom Server in sein *FMOL*-Instrument lädt, kann sich die fremde Musik sowohl anhören als auch teilweise modifizieren - z.B. einen Overdub oder Remix anfertigen und diesen gleichermaßen auf den Server laden. Dort wird das modifizierte Stück automatisch als Abkömmling des ersten erkannt und an die entsprechende Stelle einer stammbaumartigen Visualisierungsstruktur eingeordnet. So können von einem Musikstück bis zu sechs Generationen mit vielfältigen Verzweigungen entstehen.

Die Echtzeitversion von *FMOL*, die in Dresden erstmalig vorgestellt wurde, kann von zwei Spielern gleichzeitig gespielt werden. Beide sehen auf das selbe Interface und können die Spielaktionen des jeweils anderen verfolgen, auch hören beide fast das selbe, mit geringen Verzögerungen des Gegenspielers, die durch die unüberwindbaren Latenzzeiten des Netzes entstehen. Die momentane Realisierung teilt die sechs Subinstrumente des sogenannten *Bamboo*-Interfaces von *FMOL* zu gleichen Teilen auf beide Spieler auf, so dass jeder drei Klangmodule spielt. Da diese Klangmodule ebenso Modulatoren sein können, können sich die Spieler gegenseitig Klänge in ein Instrument des anderen schicken, also direkt in das Spiel des anderen einwirken. Entwickler und Musiker vernetzter Softwareinstrumente nennen diese Form des Ineinander-Hinein-Spielens »Intraaktion«.

Chris Brown präsentiert Bausteine seines Projekts *The Eternal Network Music Site*, speziell entwickelte Software-Instrumente, mit denen Musik machen mehr zum Zuhören wird, oder auch Musik hören zu einer Aktivität. Dieser Ansatz beruht darauf, konstruktiv mit dem Aspekt der Öffentlichkeit des Netzes umzugehen: Das Netz ähnelt dem öffentlichen Raum in der Stadt. Übertragen auf Musik bedeutet das u.a. hohe Fluktuation der Mitspieler,

³ *FMOL* ist Freeware und läuft auf fast allen PCs unter Windows95 oder neuer (nicht NT).

Zufälligkeit ihres Zusammentreffens, unterschiedliche Versiertheitsgrade. Brown möchte mit seinem Projekt diese spezielle Öffentlichkeit für Musik produktiv nutzen. Geplant ist, dass sich auf der *Eternal Network Music Site* Personen ohne spezielle Vorkenntnisse einloggen und auf vor-fabrizierten Instrumenten Musik hören und gleichzeitig aktiv sind.

Bei den momentan existierenden Experimentalstücken des Projektes (in Dresden wurden die drei ‚Instrumente‘ *CloudStreams*, *Invention #6* und *Bellwethers* gespielt⁴) spielen alle Spieler zusammen auf einem kollektiven System, jeder Spieler besitzt jedoch nur eine begrenzte Regelkraft. In der Dresdner Situation befanden sich alle Spieler im selben Raum, und jeder hörte gleichzeitig seinen individuellen Anteil der Musik aus den Lautsprechern an seinem Computer und den Gesamtklang der Gruppe. Nicht einmal sein eigener musikalischer Anteil des Gesamtklanges im Raum wurde vom Spieler allein gesteuert, sondern war Produkt von Duetten oder der gesamten Gruppe.

In einem Punkt ähneln sich Browns und Jordàs Konzepte: Das Spiel der Software-Instrumente ist intraaktiv, bei Brown in einem noch höherem Maße als bei Jordà, weil bei *FMOL* zwei Spieler das Interface teilen, bei Brown die Spieler jedoch wirklich mit den Aktionen der anderen interferieren und gegenseitig direkt die Resultate ihrer individuellen Spielaktionen beeinflussen. Browns Musik ist einerseits offener, weil weder Einzelklänge noch formale Entwicklungen gezielt gesteuert werden können. Der Künstler setzt andererseits bewusst engere Grenzen. Seine Instrumente sind nah an der Komposition angesiedelt: Sie arbeiten mit einem bestimmten Klangmaterial und geben sehr enge Spiel-Regeln vor. Dadurch ist die Erscheinung jedes »Stückes« charakteristisch, wenn auch die konkrete zeitliche Entwicklung nicht geregelt ist.

Guy van Belle ist digitaler Experimentalist und arbeitet mit Vielfalt. Er verwendet alle Medienformate, die der vernetzte Computer verarbeitet. Video, Audio und Text fließen in seiner Arbeit zusammen, ebenso finden sich verschiedenste Verbindungen zur Welt außerhalb des Bildschirms (also zu verschiedenen Formen von Interfaces und analogen Kontrollsignalen) in seinen Performances. Komplexe Audio-Programme wie *Max/MSP* oder die Video-Software *Nato* wie auch etliche Freeware- oder Shareware-Programme, die sich für offene Strukturen eignen, finden Verwendung. Vielfalt meint bei ihm auch die Kollaboration mit Künstlern unterschiedlicher Herkunft, d.h. die Arbeit in der Gruppe und

Download und Information unter <http://www.iaa.upf.es/~sergi/FMOL>

⁴ Alle bislang existierenden Instrumente sind in der Autorensoftware *Supercollider* geschrieben. Informationen dazu unter <http://www.audiosynth.com>

die Kreativität, die aus unterschiedlichen Sichtweisen und verschiedenen künstlerischen Methoden entsteht.

Van Belles Musik ist von der Überzeugung geprägt, dass vernetztes Musikmachen nur als bedingungslos offenes Experiment sinnvoll ist, weil die Spezifika des Mediums unweigerlich in die künstlerische Technik einfließen. Da in dem Medium Alltagskultur, Kunstpraxis, Politik und vieles mehr untrennbar verwoben sind, weist auch van Belles Musik all diese Elemente auf und spiegelt ihre komplexen Verwicklungen: Für Dresden bedeutete dies den Einsatz eigens angefertigter Max/MSP-Instrumente, die zugleich von den Workshopteilnehmern vor Ort und von fünf weiteren Musikern gespielt wurden (Akihiro Kubota, Lucas van der Velden, Gert Aertsen, Nik Gaffney und Els Opsomer in Belgien, Holland und Japan). Die externen Spieler hatten die von van Belle erhaltenen Software-Instrumente z.T. modifiziert, was bedeutet, dass hier ein musikalischer Zusammenhang vorliegt, bei dem nicht nur der Phänocode des hörbaren Resultats, sondern auch der Genocode des zugrunde liegenden Erzeugungssystems ephemere ist. Die von den Externen über Internet einlaufenden Steuerdaten waren in keiner Form überschaubar. Der musikalische Prozess war nicht mehr unter Kontrolle, höchstens leicht beeinflussbar. Einmal angestoßen nimmt diese Musik ihren eigenen Lauf. Selbst wenn es von außen betrachtet so aussieht, als würden die Spieler spielen, werden sie im Grunde "gespielt".

Alle drei Musikern behandeln die Frage, wie aus kollaborativen Strukturen und dem Prozess der medialen Vermittlung über das Netzwerk Musik entstehen kann. Brown entwirft dafür einen Container, der weitgehend festgelegte Musikstücke für eine neuartige Rezeptionsform zwischen Spielen und Hören anbietet. Jordà stellt ein Instrument mit offeneren Möglichkeiten zur Verfügung, in dem Stücke individuell improvisativ entstehen, durch die generative Entwicklungsform aber auch von Kollaborationsprozessen geprägt sind. Van Belle praktiziert absolute Offenheit der Mittel. Indem er den Genocode flexibel konzipiert und hochkomplexe Steuerstrukturen anlegt, provoziert er, dass sich seine Performances verselbständigen.

Die Künstler-Software *nebula.m81* von Netochka Nezvanova, über die ich abschließend sprechen möchte, macht einige der genannten Aspekte noch deutlicher. Das Programm spielt beliebige Dateien als Audiodateien ab (egal ob ursprünglich Text-, Bild- oder Audioformate). Diese werden automatisch von Webseiten heruntergeladen und zugleich als Kontrolldaten zur Grafiksyntaxe benutzt. HTML-Texte flackern über den Bildschirm und sind wenige Sekunden später als zerklüftetes Rauschen und Knacken zu hören bzw. als

grafische Muster zu sehen. Automatische Prozesse digitaler Signalverarbeitung mutieren, poetisieren, musikalisieren, visualisieren das Datenmaterial. Der Programmbenutzer wählt dabei zwischen verschiedenen Modi der Darstellung und der Modifikation und kann über ein nicht immer verständliches und nie genau zu steuerndes grafisches Interface Audio-Loops anlegen. Er kann diverse Effekte auf die Samples anwenden lassen, ebenso kann er eigene Audio- oder HTML-Dateien einfließen und verarbeiten lassen. Interessant wird der Umgang mit *nebula.m81* tatsächlich gerade dann, wenn man die Prozesse des Programms weitgehend gewähren lässt. Man verfolgt das meist langsame, bisweilen unerwartete Mutieren des Outputs von einem Zustand in einen anderen, greift gelegentlich vorsichtig ein, um die steuernde Hand sogleich wieder zurückzuziehen und weiter der Entwicklung zuzuhören usw.

Am Beispiel *nebula.m81* lassen sich mehrere Erkenntnisse gewinnen.

Erstens weist es Übereinstimmungen mit Browns Konzept auf: In beiden Fällen handelt es sich um Musik, die der Benutzer der Software teils hört und teils spielt. Willentliche Beherrschung von *nebula.m81* ist schwer möglich bzw. geht auf Kosten des spielerischen und hörästhetischen Genusses. *Nebula.m81* erfordert eine Rezeptionshaltung zwischen Musik Spielen (im Sinne m.o.w. virtuoser Kontrolle) und Musik Hören (im Sinne von Fließen mit einem dargebotenen Klangereignis). Man kann sogar behaupten, dass sich das gewohnte Verhältnis umkehrt: Der Spieler von *nebul.m81* wird vom Instrument gespielt, wird instrumentalisiert, ähnlich, wie es bei van Belles Musik passiert. Dies ist keine unbekannte Erfahrung bei der Auseinandersetzung mit Maschinen und mit komplexer Technologie. Beim Arbeiten mit Computern empfindet man oftmals die Maschine und nicht mehr sich selbst als Akteur, da Codesystem und Funktionsverhalten der Maschine die Interaktion, die Abläufe und die Ästhetik der Vorgänge dominieren. Insofern ist es eine naheliegende Form auch musikalischer Maschineninteraktion.

Zweitens findet man bei *nebula.m81* Ähnlichkeiten mit van Belles Vorstellung von Kreativität als Schöpfung aus möglichst großer Vielfalt und Diversität. Unsere musikalische Erfahrung bietet keine überzeugenden Gründe, dass HTML-Text interessante Musik hervorbringen kann. Genau hier setzt Nezvanova an: HTML-Text besitzt aus ihrer Sicht offenbar mehr musikalisches Innovativ-Potential als all jene Phänomene, deren Musikalität selbstverständlich ist, weil diese Selbstverständlichkeit bedeutet, dass sie musikalisch ausgeschöpft und ermüdet sind. Dem Fehler zollt sie größten Respekt, da er das auszeichnende Merkmal der höheren Organismen sei. »*All that is not information, not*

redundancy, not form and not restraints is noise, the only possible source of new patterns.«⁵

Bei Jordàs FMOL fällt eine umgekehrt proportionale Beziehung zu *nebula.m81* ins Auge. Jordà geht davon aus, dass das Interface seines Instrumentes möglichst selbsterklärend und schnell fassbar ist, damit möglichst viele Leute möglichst schnell lernen, möglichst vielseitige Musik zu produzieren. Nezvanova argumentiert mit ihrer Arbeit dagegen gerade anders herum. Sie gestaltet das Interface und die Programmfunktionen explizit kryptisch und uneindeutig. Dadurch entwickeln visuelle Erscheinung und Handhabung einen Charakter, den man als Mystik des Digitalen bezeichnen könnte. Es drängt sich ein Vergleich zu rituellen Werkzeugen auf, die den Kontakt zu übernatürlichen Welten herstellen, in denen andere Funktionsregeln herrschen als die uns bekannten. Diese werden durch das Medium ‚rituelles Werkzeug‘ zwar weder durchschaubar noch beherrschbar, jedoch ansatzweise regulierbar. Ähnlich wie mit den übernatürlichen Welten kultischer oder religiöser Prägung verhält es sich mit der fremden Welt des Computernetzwerks: Wir können nie hineingehen, und sie ist mit unserem raum-zeitlichen Weltbild nicht adäquat vorstellbar, dennoch erzeugt sie Realitäten. Weil das Regelwerk fremd und fern ist, sind diesen Realitäten neue und eigene Qualitäten eigen.

Der zentrale Punkt dabei ist, dass die Erfahrung jener Realitäten beim vernetzten Computer ausgesprochen eng an die eigene Handhabung des mediatisierenden Werkzeugs durch den Rezipienten gebunden ist. Die bloße Vorführung des Werkzeugs wirkt genauso wie die inszenierte Aufführung dieser Realität als verfälschende Reduktion. Netzmusik ist keine Darbietungsmusik.

Erscheint in: Frank Geißler (Hg.): Kunst und Künstlichkeit, Saarbrücken.

□ Golo Föllmer 2001

Kurze Textauszüge können unter Angabe der Quelle frei zitiert werden. Sollen längere Passagen wiedergegeben werden, als es für wissenschaftliche Referenzen üblich ist, bitte mein schriftliches Einverständnis erfragen unter golo@adk.de

⁵ Netochka Nezvanova: »The Internet, A Musical Instrument in Perpetual Flux«, Computer Music Journal 24:3, Fall 2000, S. 38-41. Nezvanova zitiert hier wörtlich Gregory Bateson.