Deutsche Gesellschaft für Elektroakustische Musik

Mitteilungen_14

2.9.1994

•

Auflage: 290

Die Mitteilungen erscheinen vierteljährlich jeweils Anfang März, Juni, September, Dezember.

Redaktionsschluß dieser Ausgabe: 28.8.1994

Kostenloser Versand an die Mitglieder der DegeM sowie an ausgewählte Institutionen und die Presse

Versand an andere Interessenten Elektroakustischer Musik gegen einen Unkostenbeitrag im Abonement

Redaktion: Folkmar Hein

Bankverbindung

DegeM (≈ DecimE)		Jahresbeitrag für Institutionen	200.- DM
Dresdner Bank Berlin	BLZ 100 800 00	Mitteilungsblatt-Jahresabonement	20.- DM
Konto-Nr.	05 141 941 00		
Jahresbeitrag normal	50.- DM		

Adresse der DegeM:

Deutsche Gesellschaft für
Elektroakustische Musik e.V.
Treuchtlinger Str. 8
D - 10779 Berlin

Telefon: (+49) 30 - 218 59 60
FAX: (+49) 30 - 213 98 16
Email: hein@gigant.kgw.tu-berlin.de

Zeitschriften / Bücher 2
Compact Disk´s 3
Informationen, neue DegeM - Mitglieder 3
Master in Musical Creation and Sound Technology in Barcelona 4
The Choice of Electroacoustic Music in Colombia (Juan Reyes) 5
Bemerkungen zu NICE / WEMC (Hein) 10
DegeM: Kurzinformation, Aufnahmeantrag 14
Common Music und Stella, Teil 2 (Tobias Kunze) 16
3.2 Erzeugung Musikalischer Struktur 16
3.3 Musterbildung Item streams 28
Internationaler Kalender Elektroakustischer Musik ab September 1994 41

Zeitschriften / Bücher

MusikTexte [1] 54

- *Zur Musik der Klangökologin Hildegard Westerkamp* (Vancouver), mit Werkverzeichnis
- *Der Komponist Erwin Koch-Raphael* (Bremen), mit Werkverzeichnis
- 3 Beiträge zu Roman Haubenstock-Ramati : ein Portrait von L. Knessl, ein Gespräch mit Thomas Meyer, Werkverzeichnis

MusikTexte 55

- Alvin Lucier: Gedanken zur Klanginstallation
- Julio Estrada: über den amerikanischen Komponisten Conlon Nancarrow
- Ausschreibungstext Weltmusiktage 1996 in Kopenhagen (siehe auch Kalender 31.3.1995)

Positionen 20 : Ephemere Musik (II)

- Rudolf Frisius Anmerkungen zu Prozeß- und Werkcharakter bei J. A. Riedl und anderen
- Gespräch Roman Haubenstock-Ramati mit Christian Scheib (vom Nov. 1991)
- Barbara Barthelmes: über den kanadischen Komponisten Robin Minard; sowie ein Beitrag von Robin Minard Klanginstallationen und „Nicht-übertragbare" Kunst

Mediagramm #16

- Die Musik und der Sonnenuntergang - „Irrton-Festival Berlin", Besprechung V. Straebel
- Klanglandschaften - Klaus Schöning und die Ars Acustica in Köln (Rede vom 6.11.93)

CMJ (Computer Music Journal) VOL. 18, # 2 : Composition and Performance in the 1990s—1

- Electronic Resources for Computer Music : Newsletters and electronic Publications (Music Research Digest, Leonardo Almanac, …) user Mailing lists (ISPW, CCRMA, Music Kit, Tuning, auditory organization…), Electronic Archives (CMJ, ICMA, IRCAM, CNMAT, CERL…), News Groups and electronic Bulletins (HMSL, WELL, USENET)
- Bill Schottstaedt: CLM – Music V meets Common Lisp
- Barry Truax : time shifting and transposition with a real-time Granulation technique
- Beiträge von : Leigh Landy, Larry Polansky, D. Jaffe & A. Schloss zum Thema (s.o.)
- Bericht von der ICMC 1993 (Tokyo), Conference on cognitive musicology (Finnland)

ARRAY (Communications of the ICMA) Vol. 14, #2

- 12 Seiten „Announcements" : Hardware (wie z.B. „Intelligent Music Workstation / CD-ROM) und Software (wie z.B: „MODE" von Stephen Travis Pope, neue Ausgabe von CommonMusic bei Taube - ZKM), Veranstaltungen und Computer Music Calendar 2 „Berichten (IEEE technical committee), Call for Papers, Music, … (z.B: Center for Arts and Technology New London), Stellenangebote (Uni Toronto, GRAME Lyon)
- Nachruf für Jerry Hunt, der im November 1993 in Texas verstarb; mit Discographie
- Alistair Riddell: Should computer music composers / musicians be programmers ?

Musicworks, #59 : back issue catalog (Authors index, other complex indexes), CD (ecology)

Eberhard **Hüppe**, *Ästhetische Technologie in der Musik*, Folkwang-Texte II, Band 9, Essen 1994

Peter **Hoffmann**, *Amalgam aus Kunst und Wissenschaft, naturwissenschaftliches Denken im Werk von Iannis Xenakis*, Europäische Hochschulschriften Reihe 36, Musikwissenschaft Band 110, Peter Lang Verlag Frankfurt, 1994

Trevor **Wishart**, *Audible Design, a plain and easy introduction to practical sound composition*, subscription only, incl. appendix and CD, Orpheus the Pantomime, London 1994

Martha **Brech**, *Analyse elektroakustischer Musik mit Hilfe von Sonagrammen*, Europäische Hochschulschriften Reihe 36, Musikwissenschaft Band 118, Peter Lang Verlag Frankfurt, 1994

New Computing in Musicology, Vol. 9, Center for computer assisted research in the humanities, Menlo Park, CA 94025-3443. [3]

---

[1] Adresse der Zeitschrift „MusikTexte" ist identisch mit der der GNM:
Gesellschaft für Neue Musik, c/o MusikTexte, Gladbacher Str. 23, D - 50672 Köln

[2] sind imKalender dieser Mitteilungen_14 integriert

[3] siehe Mitteilungen_13: Bestellung dieser Publikation bei Peer Sitter, Uni Köln, Email alm05@rs1.rrz.uni-koeln.de

Compact Disk´s

- Åke Parmerud, „*invisible music*“ [4] mit: Les Objets Obscures, Intermediate I - IV, Inside looking out, Alias, Exor, Jeux imaginaire. Phono sueca PSCD 72
- Ragnar Grippe, electronic music. BISCD 241
- Karlheinz Stockhausen: *Synthi-Fou, Dienstags-Abschied, Klangfarben von Synthi-Fou.* 2 CD´s #42 im Stockhausenverlag, Kürten
- INA C 2005 : Michel Redolfi
- J&W CD 931: Viviene Spiteri (Cembalo) spielt u.a. Hedman&Karlsson, Saariaho, Dolden
- Musidisc 244732 : F. Bayle *Fabulæ*
  Musidisc 244722 : Teruggi *Syrcus, Sphaera*,
  Musidisc 291302 : Ferrari, mit Ensemble Le Banquet
- WERGO 6266-2 : Genzmer, die 2 Trautoniumkonzerte mit Oskar Sala
  WERGO 6301-2 : Pierre Henry *La Ville -die Stadt*; WDR-Produktion
- MII-CD-11 : Electroacoustic Music von Joseph Dorfman und Yitzak Sadai (Israel)
- CSR-CD 8701: Barry Truax *Blind Man, Aerial, Wave edge, Solar ellipse, Riverrun*
  CSR-CD 9401: Barry Truax *Basilica, Song of songs, Sequence of later heaven, Nightwatch*

Informationen

- **Call for scores** for electric guitar and tape. Sonic arts network is calling for pieces for either electric guitar (with or without tape) or tape alone to be considered for a concert and a workshop with Tim Brady (Canada) as part of the 15th Birthday celebrations on 18th and 19th February 1995, Southbank London. Please ask for more details : snic arts network, Francis House, Francis Street, London SW1P 1DE, fax +44-71-2335159, Email Katharin Norman DORE@lse.vax.ac.uk

- **Call for EM** in Brasilia: open air concerts. Send DAT or CD or analog tapes to: Jorge Antunes; Laboratorio de Musica Eletroacustica, Universidade de Brasilia, Departamento de Musica Sala 21; 70.910-000 Brasilia DF, Brazil

7 neue Mitglieder in der DegeM:

Hildegard Westerkamp	685, W. 19th Ave.	CDN	V5Z 1W9	Vancouver, B.C.	001604	879 0731
Golo Föllmer	Johanniterstr. 10	D	10 961	Berlin	030	694 1251
Hans-Peter Kurz	Marienburger Str. 32	D	21 465	Wentorf	040	720 8667
Karl-Heinz Schöppner	Uhdeweg 18	D	22 607	Hamburg	040	899 7724
Frank Schweizer	Hohlohstr. 7	D	76 189	Karlsruhe	0721	502534
Penko Stoitschev	Postf. 304212	D	10 724	Berlin	030	217 2366
Matthias Kirschke	Potsdamer Str. 100	D	10785	Berlin	030	2623800

herzlich willkommen !

[4] zu Parmerud und seiner neuen CD siehe „svensk musik“ #2-1994

MASTER IN MUSICAL CREATION AND SOUND TECHNOLOGY

(academic year 1994-95 Universitat Pompeu Fabra, Institut Universitari de l'Audiovisual & Fundacio Phonos)

ADDRESSEES AND ADMISSION REQUIREMENTS

This Master is addressed to all persons interested in musical and multimedia creation, both in its technical and artistic aspects. After completion of these studies, the participants will be able to today's most advanced technological possibilities. To enrol, students should have an undergraduate degree in one of these areas: Music, Engineering, Multimedia, Computer Science or any other degree related with these fields. Independently of the degree, experience in music and computers is a must.

OBJECTIVES to offer

- an integrated curriculum in the fields of Music and Sound Technology.
- the needed academic foundations and practical knowledge to carry innovative productions in the multimedia and music fields.
- a musical program that fulfils today's musical needs.
- technical studies on sound of interest to the people working the areas of music and multimedia.
- To get familiarized with the most current audio technology.
- To realize a practical project, related to either research or creation.

STRUCTURE AND SCHEDULE

The Master in Musical Creation and Sound Technology is made up of 384 school lective hours, plus an undetermined number of practical training, divided into two academic years. Each academic year is divided into two semesters, the first one going from the beginning of October to the beginning of February, and the second one going from mid February to mid June. Most of the classes are accompanied by practical work to be carried at the laboratories of Phonos and Institute de l'Audiovisual, outside the times of the lectures. Students have access to the laboratories during the week and they can also use the UPF library. Enrolment is limited to 15 students.

To get the Master's degree the student has to present an original creative or research project related to one of the subjects.

Year	Semester	Subject	hours per week
94-95	First	Music theory	2
94-95	First	Computer music	2
94-95	First	Acoustics and psychoacoustics	1
94-95	First	Electronics	1
94-95	Second	Musical Analysis	2
94-95	Second	Sound synthesis and processing	2
94-95	Second	Programming techniques	2

Year	Semester	Subject	hours per week
95-96	First	History of XXth century music	2
95-96	First	MIDI and real time programming	2
95-96	First	Digital recording techniques	1
95-96	First	Music and sound in multimedia	1
95-96	Second	Composition seminar	2
95-96	Second	Computer music seminar	2
95-96	Second	Multimedia aesthetics	1
95-96	Second	Multimedia postproduction	1

FACULTY

Ana Barjau (Ph.D. in Musical acoustics)
Gabriel Brncic (composer)
Perfecto Herrera (Psycologist and technician)
Joan Josep Ordinas (musician and programmer)
Eduard Resina (composer)
Xavier Serra (Ph.D. in Computer music)
Joan Trayter (engineer in sound and image)
Coordination: Xavier Serra and Gabriel Brncic.

Pre-inscription: from September 1st to 15th. Limited admissions. It is necessary to present C.V.

Registration : from the 19th to the 30th of September

Fee : 500.000 PTA for the two academic years

Information and Registration : Institut Universitari de l'Audiovisual
La Rambla, 31, E - 08002 Barcelona
tel.: +34 - 3 - 412 3991
fax.: +34 - 3 - 412 4162

Juan Reyes

Department of Music, Universidad de los Andes
Santafé de Bogota, Colombia
Email: juanig@cup.portal.com

The Choice of Electroacoustic Music in Colombia

Electroacoustic music in Colombian music compositions appeared as early as in the 1960's as part of experimental works, sound tests, and homework for some outstanding composers in the country. It was not until 1968, when electronic sounds were accepted as an art form and used in few compositions by a handful of artists and composers, namely Jacqueline Nova (1935-1975) et al. After her death most Colombian composers and musicologists regarded this medium as an isolated event with few consequences among the musical community. Most activities were restricted to broadcasting classic electroacoustic compositions by radio stations mainly in Bogota.

It was not until 1989 during the International Contemporary Music Festival, when few composers and musicians who have been active during the "Nova" years decided to dynamically support and revive electroacoustic from its static state it had for years. Since then there have been various activities promoting this kind of music which range from compositions to workshops, acousmatic cycles to research and festivals.

In the last five years the electroacoustic genre has proven to be more than a tool, to be a medium of expression for composers and listeners as well. For the porpoise of better understanding this changing of view on the musical community of today, a look at few points on the view of the history of music in Latin American Countries, would not be insufficient.

In contrast with Latin America music, European music is logic, Its history is part of a series of continuous development and research in all of its technique, and aesthetics. The differences in musical thought can be seen in the trade mark or better originality on its works. Each geographical location has been devoted to a particular genre, and has succeeded among the history of western music.

There has been a logical sequence of techniques, that have been applied to European instrument design and compositions to be played on such instruments. There is organised thought in most music schools and conservatories. Research is continuous and methodical. The consequences of all this planned organisation are a series of coherent and clear facts that appeal to the ears of educated audiences as well as players and composers worldwide.

Innovation of European thought depends upon the need to better communicate a musical feature based on the spirit of an age. This means that musical enlightenment is gradual and a function of an exchange of ideas between composers and theoreticians. This, allows for showing and proving new concepts as well as those of a heritage of classical music compositions of western music through the lives of its creators. The spirit of the European musicians depends on the future of their tradition.

Latin American Music instead, is not a function of its own values and cultural facts. This exists as various emotional impulses ranging from mind maturity or growing, mind states, cultural and racial mixtures, and trans.-cultural values. As a result, a phenomenon like this, leaves the musicologist with almost no method to apply western thought to analyse questions on music of the central and southern parts of the American continent.

In comparing the works of an outstanding European composer like Arnold Schönberg with those of a great South American Composer, Heitor Villa-Lobos, we see that, while the works of the second composer are part of a surprise and spontaneous phenomenon, the works of Schönberg are part of a family of intellectual, and musical heritage. The compositions of Villa-Lobos are consequence of discontinuous struggle to adapt western music techniques to his compositions and work

style. atonality is part of a historical development in music taking place in central Europe. The compositions of the Brazilian composer are an amalgam of the most advanced methods used in all the countries of the old continent.

Inherent to the nature of Latin American music, there always have been a nationalistic current rooted on the basis of folklore on each country. Depending the region, in most cases, folklore, is a mixture of American Indian traditions and Spanish traditions. On the Caribbean and Brazil, black or African or Negro traditions might be added.

This nationalistic way of composing is due in part to migration of European and North American composers to Hispanic countries of the continent. Being more specific, since the days of colonies, music education has always been a continuous emulation of those methods proven to be successful in the northern hemisphere. This is due in part by the travelling of a selected group of Creole musicians to renown music schools and centers around the world providing a "westernation" of Latin music.

With colonization, came missionaries and with them not only religious activities but also cultural activities which always included music. About this time, they established a set of music priorities, which had to be used by the music educator to characterize a musical expression on the colonies. In order of importance they are, vernacular, semi-vernacular, popular, dance music and folklore. The last one was only used as source of inspiration, for the others. Many pieces were done by using this set of priorities. As good as they could be, they today remain in the archives of the conservatories where they were composed, and only a few have made its way to the history of universal music.

Most composers have made use of instruments of traditional orchestra, and most musicians have always performed on this type of instruments. Nevertheless the guitar brought to this continent by the Spaniards, is by far the most popular instrument. One can easily find variations to this instrument in every region of the continent. And a great deal of compositions for this family of instruments exists in every category, but most composers still favored those of the orchestra.

There have been other composers who have used native instruments or Indian instruments which include, pan flutes of many kinds, stone and clay flutes, whistles and seashells, as well as huge family of drums inherited by the age of the Spanish black slavery on the Americas. But the general practice of vernacular music has pointed towards the use of emulating techniques to adapt sounds of these native instruments to the sounds of strings, kettle drums, woodwinds, and brass. The product of these practice are symphonies based on local dances which clone custom instruments with classic harmonies, rhythms and orchestrations.

In music schools around all Latin America the rules which have prevailed the most are those used in traditional western compositions. We see waltzes, marches, sarabandes, fugues, sonatines tone rows and others, on the works of important composers in every country. In contrast to this fact, some European and North American composers who have traveled by these lands have composed orchestral music using tangos, sambas, rancheras, cumbias, rumbas and bambucos. With the aid of contemporary thought on music, the music of today allows this regional or native instruments to be part of the orchestra. In this way, dances and folklore are increasingly being part of many new compositions world-wide. Today we see how composers are including Andean, Indian, and Caribbean instruments on their compositions of traditional acoustic or electroacoustic music.

If as people from the central to the southern part of the new continent, we look for a substantial contribution to the history of music of mankind, we have to look for a collection of new priorities on how music is conceived, composed, produced, and performed. For people on this side of the world, music is inherent to every activity. By nature the Latin has few preconceptions about what music should be, and therefore performs music in a primitive but spontaneous way. Since it is difficult to rationalize the way every individual performs and the way a sound is produced, a study of several cases could possibly give some clues which in place could help or influence the way an instrument

or a voice are performed. New compositions do not have to elaborate on sophisticated compositional techniques, they do have to focus on the freedom of expression, qualified by the state of mind of the cultural surroundings in the creative mind. Sound sources could be explored by getting a new blending of tone colours either with traditional instruments or by methods of subtractive synthesis. In any case we return to the fact that most of the music of Latin composers is a realisation of dreams and imaginative processes which might as well started to be modeled in modern virtual worlds.

Most of the history of Latin America can be applied to every country and geographical region ranging from Mexico in southern North America, Central America, and South America. Colombia is on the northern part of South America. Since it is surrounded by the Atlantic and Pacific oceans, it has a mixture of musical heritage found in Central American and the Caribbean it also has the Hispanic traditions brought by the roads built by conquistadors, from the Atlantic to the golden gods of Peru in the middle of South America. The Andean mountain range spreads into three cordilleras or smaller mountain ranges leaving also many Andean traditions.

Turning back to our subject of electroacoustic music in Colombia, It can be seen that for historical, and traditional matters, most audiences have been isolated from this genre. Nevertheless as stated earlier on this article, only a handful of composers paid attention on composing using electronics.

Ensayo Electrónico was the first electroacoustic piece to be composed in Colombia in 1965 by Fabio Gonzalez-Zuleta (b. 1920) and technician Guillermo Diaz. The piece was produced and recorded at the Colombian National Radio Broadcasting studios in Bogota. It used tone generators and the tape manipulation techniques of the Cologne studio. Another prominent figure of contemporary music in the country Blas Aterthua (b. 1933), and during a workshop in Buenos Aires, Argentina composed *Sirigma* (1966) which is a piece for tape using recorded piano sounds, processed also by basic tape manipulation techniques. In 1968 Jacqueline Nova (1935-1975), produces *Oposición-Fusión*. This piece was also realised during a residency at Estudio de Fonología in Buenos Aires.

The most important electroacoustic piece during the 1970's was *Cantos de la Creación de la Tierra* (1972), also by Nova. This piece was composed by using tape manipulation techniques, wave distortion techniques, as well as filtering on voices of Colombian Indians. In 1975, Francisco Zumaque (dates N/A) composed his first electronic etude called *Mexcalito*. This is a composition for traditional instruments, live electronics and tape.

Musicologist Carlos Barreiro-Ortiz, since the 1980's, has produced and directed a series of contemporary music radio shows. These in many occasions have included classic electroacoustic works as well as Colombian electronic music compositions. Some of the works included compositions produced in The United States using tape or live electronics by Colombians Guillermo Gaviria, and Andrés Posada (b. 1954). Mr. Barreiro since then have been the only promoter of this music in the country by producing not only the show but also a series of concerts with electroacoustics as its subject matter.

With the death of Jacqueline Nova in June of 1975, almost no trained Colombian composer explored electroacoustic territories until the First Contemporary Music Festival of Bogota directed by Nova's student and assistant Cecilia Casas in 1989. At that time the music of Karlheinz Stockhausen, John Cage, Kaija Saariaho, and others was performed for the first time on various auditories in Bogota. At the same time, the first real-time computer music recital takes place with the composition *Música por Computador para el oyente desprevenido* (1988) by Juan Reyes (b. 1962).

In 1990, the first studio devoted to electronic music and computers was founded by engineer Camilo Rueda and composer Andrés Posada with the assistance of engineer and composer Francisco Iovino. The center was named after Jacqueline Nova and is now on the hands of composer Fabio Fuentes (b. 1957). *Caternaria* by Posada was the first piece composed on the studio. In December of the same year the first electroacoustic cycle takes place on the Colombian-American Center. It

featured tape pieces of composers from the continent as well as some pieces from IRCAM. It also featured *Dialogos por Paz* (1990) by Juan Reyes. This piece was composed using tape sounds from the assassination of political leader Luis Galán.

In 1991 the second contemporary music festival took place with no much emphasis on electroacoustic music. Still, he Festival highlighted works of Messias Maiguashca, and Javier Alvarez. The Compositions *Anagramas* by Juan Reyes and Colombian composer and architect Mauricio Bejarano (b. 1955) were premiered at this time. Its recording which is the first of electroacoustic music to be released in the country was due for 1992. Also during this festival, Colombian live-electronics musicians and composers Roberto Garcia (b. 1958) and Ricardo Arias (b. 1964) residing in Barcelona Spain, perform with their ensemble "Sol Sonoro" for the first time in Colombia. *Monológo para la ciudad muerta* for flute, double bass, percussion and electronics, and *Requiem para una muerte anunciada,* for flute and live electronics by Catalina Peralta (b. 1961) were also premiered during 1992. *Cordales* for String instruments and tape by Guillermo Carbó (b. 1963) are also first heard in the country at that time. *Mimetismo* by Canadian acousmatic composer Stephane Roy was performed on August, 1994 by Colombian guitarist Arturo Parra. The work was commissioned and dedicated to the guitarist.

1993, is the year when the first acousmatic cycle takes place, as part of the third Contemporary music festival. It was directed and produced by Mauricio Bejarano and Juan Reyes with the help of the Colombian Culture Institute and the Colombian Science Office. The works featured at that time included pieces by Francis Dhomont, and composers of the University of Montreal, Larry Austin, and several composers of the International computer music association, and Paul Lansky. Daniel Teruggi of INA-GRM who was present at the cycle performed pieces of composers from this center. Jon Appleton from Dartmouth College in the U.S. performed with the radio baton for the first time in Colombia.

After the cycle, electroacoustic music was better accepted and supported among the academic circles of music departments at the National University of Colombia, and at the University of Los Andes. For the first time electroacoustic music history and compositional techniques were added to the curriculums of both universities. Also at that moment, The Colombian Electroacoustic Composers Association was formed with members in the country and Colombians abroad.

On the first half of 1994, The first Electroacoustic music competition takes place. The entries included works of students from few universities from around the country. By mid 1994 *Ecos y Resonancias* by Juan Reyes are the first computer music compositions to be used on the stage by a Colombian drama and theater company. These pieces were continuously performed for more than a month.

Electroacoustic techniques used up to these days by Colombian composers include, tape manipulation techniques. Wave distortion techniques like amplitude modulation, ring modulation, and frequency modulation. Filtering and subtractive synthesis. Additive synthesis by using tape techniques. Live electronics with real time digital signal processing and tape processing. Computer music technologies like digital synthesis, phase vocoder, filtering, and environment simulation. And also algorithmic composition.

The reasons for a Colombian to pursue a career in electroacoustically methods of compositions in these days abound. This music offers unorthodox compositional techniques which might be adapted to the way composers work in Colombia. Musical heritage in Latin America can be researched by using mere sound characteristics and sound production techniques, therefore avoiding tonality as the basic compositional tool. Composers who are not trained in the traditional conservatory or western way of thinking, can still create music of some significance provided by the rules of an aesthetic native style.

Since diatonic ear training is not the basis for perceiving music, audiences who have not previously perceived music in a rational way, might be more open to electroacoustic sounds. This means that the cultural aspects of Colombian musical heritage might be applied directly to an art form, instead of adapting them to instrument and orchestrations of other latitudes. Contemporary music can be composed independently from the medium. Therefore a composer do not have to rely on an orchestra to express his creative works. The work can be done interactively by listening to the composition in an immediate way. This leaves the composer free on a direct two way relationship with the piece being created.

The music of today does not have to be created specifically for the concert hall. In relation with the other arts, the techniques of fine arts can be applied to sound in electroacoustic music composition. Music can be seen from the scope other arts much easier since it has less abstract qualities than traditional common practice music. New forms like art-music, sound installations, performances, happenings, and dance can be created from electroacoustic music as its basis because it can easily blend with the other electronic arts.

With the exchange of information, developments in other countries get spread relatively faster. Therefore there is a better communication of ideas among composers, theoreticians and researchers around the world. In addition pop technology is advertised and applied around the world in the same manner. This makes possible to communicate around the world with almost the same language.

For the newcomer, the electroacoustic medium is more reachable than the orchestral medium because electroacoustic music performed in Colombia is more feasible than traditional orchestral music. With the electronic infrastructure of the country compositions can be transferred to many formats, and consequently can be broadcast and distributed, among more individuals than those in a concert hall.

Since music in most Latin American countries continues to be a phenomenon or magic, It could be that electroacoustic music be regarded shortly as an art form with no limits and borders. Thus, its aesthetic qualities will be inherent to its own principles and will depend only on those individuals who are working on this kind of music rather than a geographical location or cultural heritage.

References

Introduction To Electroacoustic Music, Schrader Barry, Prentice Hall, Englewood Cliffs, NJ, U.S.A., 82

Computer Music: Synthesis Composition and Performance, Dodge Charles, Jerse Tom A., Schirmer Books, New York, 85

Composers and the Computer, Roads Curtis ed., A-R, Los Altos, CA, U.S.A., 91

America Latina en su música, Aretz Isabel ed., Unesco, Siglo Veintiuno editores, Mexico DF, Mexico, 77

Historia de La Música en Colombia, Perdomo Escobar Jose Ignacio, Editorial ABC, Bogotá, 63, third edition

En Torno al Barroco, Obras de Fabio Gonzalez Zuleta, Barreiro Ortiz Carlos ed., Centro Colombo Americano, Bogotá, 86

A proposito de Jaqueline Nova, Barreiro Ortiz Carlos ed., Centro Colombo Americano, Bogotá, 83

Música Electroacústica Sala Tairona, Barreiro Ortiz Carlos ed., Centro Colombo Americano, Bogotá, 86

Jovenes Compositores Colombianos, primera a septima audiciones, Barreiro Ortiz Carlos ed., Centro Colombo Americano, Bogotá, 87 - 93

Bemerkungen zu : **NICE / WEMC**

Nach dem Austritt der DecimE aus der CIME (Beschluß auf der Jahresversammlung 1994) wird im folgenden darüber informiert, wie sich seither die Diskussion um die Gründung einer weltweiten Gesellschaft für EM weiterentwickelt hat. Anschließend (in Punkt 4 und 5) können Sie eine Stellungnahme der Redaktion lesen, die nicht unbedingt mit der Meinung des DegeM-Vorstandes in allen Punkten übereinstimmen muß.

Der Leser möge sich die Ausführungen in Ruhe durch den Kopf gehen lassen und gegebenenfalls der Redaktion seine Meinung und Vorschläge mitteilen, um so zur öffentlichen Diskussion beizutragen. Spätestens auf der nächsten Jahresversammlung der DegeM Ende Juni 1995 in Essen wird dieses Thema aufgegriffen und weiter diskutiert.

1. Ein erster Ansatz der „Neuschöpfung" steht in engem Zusammenhang mit den Absichten, wie sie im Madrid-Dossier 1992 (gemeinsam verfaßt von den „Aspiranten" sonic arts, PEM, GEM und DecimE) in Opposition zur CIME zum Ausdruck kamen und danach von Konrad Boehmer (PEM) in einer Fragebogenaktion [5], die die Vorstände auch beantwortet haben, grundsätzlich definiert wurden und nun zur Weiterführung anstehen. Zunächst sollen 3 Personen von den Aspiranten ernannt werden, die eine Satzung zur Diskussion und späteren Verabschiedung erarbeiten . Die bisher interessierten 6 Aspiranten sind: GEM (Österreich), CEC (Kanada), DegeM (Deutschland), Ars Sonora (Frankreich), PEM (Holland) und sonic arts network (UK). Bisher hat man, kurzgefaßt, folgende Prämissen angedacht:

Es wird ein (zunächst) aus 6 nationalen Mitglieds-Organisationen bestehender Verein ‚ohne Zentrum' (ohne Büro) mit dem Namen „**NICE**" [6] gegründet, der einen 5-köpfigem Vorstand (falls finanzierbar: mit einem zusätzlichen Geschäftsführer) und eine noch zu entwerfende Satzung hat, welche in einem Land mit liberaler Gesetzgebung zugelassen ist. Die eigentlichen Entscheidungen werden - dieses Procedere ist noch nicht klar - auf einer alle 2 oder 3 Jahre einzuberufenden Versammlung getroffen. NICE finanziert sich aus staatlichen Zuschüssen bzw. aus Mitgliedsbeiträgen. Das wichtigste Ziel von NICE ist vorläufig so definiert: Informations- und Produktionsaustausch zwischen den Mitgliedern; darüber hinaus soll NICE die Mitgliedschaft in der UNESCO, im Internationalen Musikrat sowie anderen internationalen Organisationen anstreben. Es gibt einige praktische Vorschläge: die DegeM könnte das Informationsbulletin inklusive. Kalender herausgeben; man strebt auch eine engere Zusammenarbeit der Studios im internationalen Rahmen an, was auch ein elektroakustisches Netzwerk erfordert.

2. Ein zu **1.** erweiterter Vorschlag der GEM (Igor Lintz-Maues) sieht zunächst einen anderen Namen als NICE, nämlich „**WEMC**" [7] vor. Die aus den gleichen Aspiranten bestehende lockere Interessengemeinschaft (kein Verein) wird von den jeweiligen Vorsitzenden der Aspiranten - Vorstände geleitet. Von den Aspiranten entsandte Repräsentanten bilden die verantwortliche Mitgliederversammlung (jeder Aspirant schickt einen Vertreter = eine Stimme). Entsprechend dem Wunsch laut Fragebogen (s.o.) soll der „Sitz" der WEMC sozusagen gleichmäßig verteilt werden auf die Sitze der Aspiranten. WEMC korrespondiert über ein Kommunikationsnetz, um schnelle Entscheidungen einstimmig herbeizuführen. Es wird vorgeschlagen, daß die Aspiranten in Ausübung ihrer Arbeit und Funktion verpflichtet sind, das Markenzeichen „WEMC" zu verwenden und stets anzumerken.

Zur Gründung der WEMC wird von der GEM folgendes Verfahren vorgeschlagen: jeder der Aspiranten schreibt jeweils an die anderen Aspiranten eine Einverständniserklärung. Der von der GEM vorgeschlagene Text lautet: „We agreed with the creation of a WORLD

[5] Type of organisation, how should it structured, ruled, function, financed, what are the aims ?
[6] New International Community of Electroacoustic Music
[7] The World Electroacoustic Music Community

ELECTROACOUSTIC MUSIC COMMUNITY having the structure delineate at the letter from the GEM-Austria from June 24, 1994; and we want to joint it“. Falls der vorgeschriebene Text nicht akzeptiert wird, kann ein anderer geschrieben werden - allerdings ist „das Spielchen“ erst dann beendet, wenn die Einverständniserklärungen aller Aspiranten identisch sind (d.h. wenn man sich auf einen gleichlautenden Text geeinigt hat). Dieses Verfahren zur WEMC-Gründung mit dem Ziel der eigentlichen Arbeitsaufnahme (bis dahin ist ja noch nichts passiert) wird von der GEM als Plattform für das weitere Vorgehen angesehen.

3. Nach der zurückhaltenden Antwort der DegeM auf diesen GEM-Vorschlag schreibt Igor in der Email vom 1. Juli 1994: [8] „Wir sind der Meinung, daß wir im Moment keinen neuen internationalen Verein (mit Vorstandswahl, Statuten, usw.) brauchen. Das schließt nicht eine zukünftige Entwicklung in diese Richtung ein [9]. Wir möchten über den Schwerpunkt ‚Kommunikation und Austausch auf internationale Ebene‘ weiter diskutieren und glauben, daß eine informelle elektroakustische Gemeinde als erster Schritt genügen würde“.

Diese im GEM-Vorschlag nicht explizit ausgedrückte Einschätzung, nämlich die Absage an eine internationale Gesellschaft und die Gründung einer eher losen „Community“, hatte bereits am 29.3.94 Daniel Leduc (CEC Kanada) viel ausführlicher in einem Schreiben an Konrad Boehmer geäußert, nämlich:

„I read with interest vour fax in response to my letter to our Austrian friends concerning the never ending CIME case. I would like to take this opportunitv to propose an idea to the electroacoustic milieu regarding our unsatisfied interests (partly due to CIME's inertia) in communication and exchange.You ask me to present ideas concerning the creation of a new world-wide association, but I will not, for I don't think we want a second association, a second CIME. It's a question of physical and structural order. In the days of the global village, the idea of an association seems obsolete. It is the reason why I will tell you about the urgent need of a new world -wide electro-acoustic community, a channel of exchange and discussion, with words and sounds from individuals, associations or groups of individuals“. Im Brief folgt eine Erläuterung des Electronic Highways mit dem Internet einschließlich des Transfers von Audiofiles; Leduc schlägt eine Newsgroup vor, speziell die Installation einer entsprechenden Diskussionsgruppe sowie eines Audiofile-Pools. Über dieses internationale Netzwerk glaubt er „that there is enough individuals and composers to feed this international network easily, quickly and cheaply. All we need is a server and a decent amount of memory to keep this information available for a while. National associations should also consider giving their members access to the Net, perhaps only from their offices. This world-wide virtual electroacoustic community would be a platform and a democratic market, active and privileged, offering an infinite and constantly changing panorama. A pool of communications and works offered by us, for us and to others interested in this music“. Es folgt noch der Vorschlag, im CEC-Bulletin „Contact!“ Meinungen zu diesem Thema abzudrucken und über die Mailingliste der CEC zur allgemeinen Diskussion zu stellen.

Zu diesem Brief kommentierte Boehmer, daß nach Rücksprache mit Ars Sonora und PEM eher eine strukturierte Organisation erwünscht sei und - wohl hinsichtlich des elektronischen Highways - daß man sich sehen müßte, daß der direkte Kontakt zwischen den Menschen unverzichtbar sei!

4. Meine Schlüsse und Folgerungen

Zur Vorgehensweise:

Der GEM-Vorschlag zur Email-rollierenden Einverständniserklärung scheint mir so lange unsinnig, wie wir „Aspiranten“ noch nicht alle von der Gründung einer internationalen Gesellschaft für EM überzeugt sind. Eine lockere EM-Kommunikationsgemeinde muß nicht gegründet werden, weil

[8] die hier aufgeführten Zitate sind original wiedergegeben (wenige Rechtschreibfehler korrigiert).
[9] gemeint ist vermutlich „schließt nicht… aus“.

sie entweder von selbst entsteht (jemand übernimmt die Initiative) oder schon existiert (Netz-Gemeinden wie z.B. die ICMA, die IRCAM-User-Group, die vielen Newsgroups).

Ich meine, daß eine internationale Gesellschaft gegründet werden soll - so richtig schön mit Vorstand, Satzung und Mitgliederversammlung (Begründung siehe unten). Es geht jetzt um die Formulierung der Satzung (worin auch die Ziele benannt werden). Was das Kürzel WEMC angeht, ziehe ich NICE vor - eben weil es ganz nice und in der Wortkonstruktion irgendwie genial ist.

Zum internationalen Netzwerk:

In der Tat existiert das Internet und vermutlich schon bald werden auch soundfiles über Internet transportiert werden. Die Frage wäre: wozu, was ist das Ziel dieser Massenmobilität der Bits und Samples? Ist die Ansammlung von Daten überhaupt noch anzustreben, wo wir doch genau wissen, daß sie zum größten Teil ungesichtet bleiben? Die Frage wäre auch: müssen wir etwas einrichten, was ohnehin schon da ist bzw. im Ausbau begriffen ist? Institute wie das IRCAM, Mills, CCRMA, MIT sind international ausgerichtet und zugänglich, sie tun bereits das, was angesprochen ist und man kann dies jetzt und mit geringerem Aufwand nutzen.

Ist es nicht leichtsinnig zu behaupten, Internet sei „einfach, schnell und billig, man brauche nur einen Server“? Wir wissen doch alle, daß die Nutzung des Internet nicht so ganz einfach ist, weil allein von unseren 129 DegeM-Mitgliedern nur etwa 16 „Privilegierte“ eine Email-Kennung besitzen [10] und weil der Transfer von größeren Datenmengen über Internet zu erheblichen Schwierigkeiten führt (es geht bei bisher realisierten Projekten um lächerlich geringe Kapazitäten von nur wenigen Sekunden). Audiodatentransfer und vor allem -speicherung und -archivierung sind ja schon innerhalb der Studios problematisch: wie lange dauert die bloße Kopie über Ethernet von 12-Minuten Stereo ? welche Studios besitzen mehr als 10 GB und haben noch x GB für den Netzservice frei? An welche Kapazität wird überhaupt gedacht: sind es 1 Minute (10 MB), 1 Stunde (600 MB), 1Monat (≈ 420 GB)? Unsere angestrebte, vorläufig schwache Organisation bzw. lockere Usergroup wird doch restlos überfordert sein, wenn sie dieses Kommunikationsziel in Eigenregie erreichen will!

Leducs Einschätzung der Kosten, nämlich die Nutzung des Internet sei billig, ist überhaupt nicht klar, denn die Zeiten der „unbezahlten“ Internet-Nutzung sind gezählt und „umsonst“ ist der ganze Betrieb in Wirklichkeit nicht. Wäre noch die Einrichtung des Servers zu bewerten: wer stellt das Personal bereit und wer bezahlt das alles? Ich denke: die DegeM hat dafür kein Geld. Wir müssen dies den Institutionen selbst überlassen und die Gesellschaft herauslassen – wie gehabt.

Was unsere eigenen Belange angeht, scheint mir weiterhin der Schritt in Richtung DegeM-Archiv beim IDEAMA des ZKM relevant zu bleiben (denn mehrere Aktivitäten gleichzeitig können wir uns einfach nicht leisten) - erst wenn wir das geschafft haben, kann man an größere Aktionen denken.

Zum Inhalt:

Bei der Einrichtung einer internationalen Kommunikationsgemeinschaft wird beschworen, daß sie selbst die Verbreitung und Aufführung von EM aus aller Welt in alle Welt auslöst bzw. ermöglicht. Dies ist durchaus unser wichtigstes Ziel! Wer präsentiert aber in Konzerten oder Festivals das internationale Repertoire von EM? Es sind die **Veranstalter**. Wer sind diese Veranstalter? Es sind in Deutschland überwiegend Institutionen, die von der öffentlichen Hand, also von den Ländern, Städten, Gemeinden und Regionalverbänden, abhängig sind. Diese Veranstalter-Institutionen sind

[10] in Deutschland verfügen wahrscheinlich nur die Studios von ZKM und TU-Berlin über eigene Internetknoten des Ethernet o.ä., die anderen nutzen Dienste der Unis (Köln, Essen, Bremen) oder etwa des GMD. Diese Frage sollte jetzt umfassend recherchiert werden!

übrigens die typischen juristischen Mitglieder der DegeM [11] . Bei der Gründung der DecimE wurde diskutiert, in wie weit sie selbst als Veranstalter auftritt [12]. In unsere Satzung wurde eingearbeitet: keine eigene Veranstaltertätigkeit der DecimE bzw. der DegeM, aber Förderung und Koordination im Dienste der Veranstalter.

Das gleiche gilt auch für den internationalen Verein NICE. Aber es scheint da eine Ausnahme zu geben. Den einzigen hier relevanten Tatbestand eines internationalen Veranstalters, der auch Verein ist, stellt nämlich die IGNM mit ihren Weltmusiktagen dar. Das Geld für die Weltmusiktage kommt jedoch nicht von der IGNM selbst, sondern von den örtlichen Veranstaltern (noch nicht einmal von den nationalen Mitglieds-Sektionen) und damit direkt oder indirekt wieder aus der öffentlichen Hand der Ausrichterstaaten. Abgesehen davon, daß man allzu deutlich die Handschrift der ausrichtenden nationalen Veranstalter spürt, die vor allem die Präsentation ihres Eingemachten im Sinne haben, erzielt die Ausrichtung der Weltmusiktage jedesmal in einem anderen Land von alleine schillernde Facetten.

Bei der Gründung der CIME, die ja Ausgangspunkt dieses ganzen Diskurses ist, wollte man (so hat es Clozier selbst 1982 in Graz ausgeführt) ein Pendant zu den Weltmusiktagen entwickeln [13] mit ausdrücklichem Anspruch auf Veranstalterstatus - nur ist dieser resultierende Veranstaltungsbruder, das jährliche Festival Synthese in Bourges nämlich, mit genau einer festgeschriebenen Programmkommission an genau einem Ort festgenagelt, was zu einer total einseitigen Ausrichtung des internationalen Gedankens führte und zu Recht auf heftige nationale und internationale Kritik gestoßen ist. Aus dieser Erfahrung heraus sollten wir für NICE diesen verheerenden Veranstalterstatus meiden. Es wäre ausreichend, sinnvoll und gut, sich auf die internationalen bzw. nationalen Informations- und Koordinierungsdienste zu beschränken, zumal EM am ehesten unter den Musikgattungen einen internationalen und kommunikativen Charakter bewiesen hat.

5. Ziele und Inhalte:

Als Ziele und Inhalte der NICE verbleiben aus besagten Gründen weder die Einrichtung eines Kommunikationsnetztes noch die Errichtung eines Veranstalterstatus – wie von Anfang an beschieden – sondern:

- Informations- und Koordinierungs**-Dienste** (zunehmend in elektronischer Form)
- Herausgabe eines von allen getragenen und unterstützten internationalen **Bulletins** (zunehmend in elektronischer Form – s.o.), das einen umfassenden Veranstaltungskalender, Hinweise auf Aus- und Weiterbildung, Kurse, Wettbewerbe, Interpreten, Adressen, Veröffentlichungen (Tonträger, Bücher, elektronische Medien) und unabhängige Produktinformationen (Hard- und Software) sowie wissenschaftliche Berichte etc. enthält
- **Mitsprache** bei der Verteilung von internationalen Mitteln (UNESCO, EU), Mitgestaltung in der Kultur- (Musikräte, IGNM) und Medienpolitik (CISAC - GEMA, EBU) …

Folkmar Hein

---

[11] z.B. Klangart Osnabrück, ZKM, die Studios Folkwang Essen, HfM Freiburg und TU Berlin; die Veranstalter HfM Köln, Stuttgart usw. sind leider nur durch natürliche Mitglieder vertreten.

[12] die DDR-Sektion der CIME - aus der die DecimE hervorgegangen ist - nahm für sich Veranstaltertätigkeit wahr, man kann sogar sagen, daß dies ihre Hauptmotivation war

[13] weil den Weltmusiktagen nachgesagt wurde, sich nicht genügend für die EM einzusetzen. Es hat sich aber in den letzten Jahren gezeigt, daß die schwerfälligen IGNM-Weltmusiktage je nach Einstellung der örtlichen Veranstalter zur EM durchaus willens und in der Lage waren, EM aufzuführen - siehe auch selektierte Werke für die Weltmusiktage bzw. Ausschreibungstexte hier im Heft

DEUTSCHE GESELLSCHAFT FÜR ELEKTROAKUSTISCHE MUSIK E. V. (DegeM)

Die „Deutsche Gesellschaft für Elektroakustische Musik“ (**DegeM**) ist Mitglied im Deutschen Musikrat und in der GNM. Sie wurde am 26. April 1991 als „DecimE“ (Deutsche Sektion der CIME ["*Conféderation Internationale de Musique Electroacoustique*"]) in Berlin gegründet

Die DegeM fördert die elektroakustische Musik in nationalem und internationalem Rahmen. Diesem Zweck dienen die Organisation von Fachtagungen, -kursen und Konzerten, der internationale Austausch von Informationen sowie die Herausgabe von Publikationen und Tonträgern. Insbesondere wurde ein Archiv in Zusammenarbeit mit dem ZKM Karlsruhe aufgebaut, in dem in Deutschland entstandene bzw. erdachte Produktionen Elektroakustischer Musik erstmals gesammelt und öffentlich zugänglich gemacht werden.

Bisher erschienen folgende Publikationen:

- "***Dokumentation*** *Elektroakustischer Musik in Europa*" (8700 Werke, 260 Studios; Index, 370 S.). Die Datenbank enthält Mitte 1994 bereits 13000 Werke (Basis für das Archiv [s.o.])
- "*Die* ***Analyse*** *elektroakustischer Musik - eine Herausforderung an die Musikwissenschaft?*". Beiträge von Klaus. Ebbeke, Gottfried Michael Koenig, Elena Ungeheuer, Dirk Reith, Kai-Erik Ziegenrücker, André Ruschkowski, Jürg Stenzl und Thomas Nagel.
- Vierteljährliche ***Mitteilungen*** mit Informationen aus allen Bereichen der elektroakustischen Musik einschließlich eines internationalen Veranstaltungskalenders. Die bis Juli 1994 herausgegebenen 13 Blätter wurden an die Mitglieder und Abonnenten sowie an international wichtige Informations-Zentren und Institutionen verschickt.
  Auflage zur Zeit: 280.
- **CD** mit 6 Produktionen des Studios der Akademie der Künste zu Berlin (1992).

Die DegeM ist selbstlos tätig und verfolgt ausschließlich gemeinnützige Zwecke. Sie finanziert sich hauptsächlich aus Mitgliedsbeiträgen und Spenden.

Aufnahme in die DegeM können Personen und Institutionen beantragen, insbesondere Komponisten, Musikwissenschaftler, Tonmeister und Tontechniker, Interpreten, Ensembles, Studios sowie entsprechende Institutionen und Veranstalter aus dem In- und Ausland. Damit sollen alle Menschen erreicht werden, die elektroakustische Musik komponieren, interpretieren, lehren, lernen, erforschen, aufführen, organisieren und verbreiten.

Die DegeM hat gegenwärtig 129 Mitglieder, unter denen sich 7 Institutionen befinden.

*

Der Vorstand der DegeM:

Folkmar Hein (Vorsitzender, Berlin)
Thomas Gerwin (Karlsruhe)
Prof. Dirk Reith (Essen)
Dr. André Ruschkowski (Salzburg)
Tobias Kunze (Berlin)

Deutsche Gesellschaft für Elektroakustische Musik

Treuchtlinger Str. 8

D - 10 779 Berlin

Tel.	+49-30 -	218 59 60
	+49-30 -	314 22821
FAX.	+49-30 -	213 98 16

Email: hein@gigant.kgw.tu-berlin.de

❑ Ich beantrage die **Mitgliedschaft** in der Deutschen Gesellschaft für Elektroakustische Musik (und erhalte automatisch die Mitteilungen)

❑ Ich möchte **Abonnent** der Mitteilungen und Publikationen der Deutschen Gesellschaft für Elektroakustische Musik werden

❑ Ich bitte um die unentgeltliche Zusendung der Mitteilungen und Publikationen der Deutschen Gesellschaft für Elektroakustische Musik; ich arbeite für

_____

(bitte Institution angeben)

❑ Ich **überweise** für das Kalenderjahr 1994 :

50.- DM Jahresmitgliedsbeitrag als natürliche Person ❑

200.- DM Jahresmitgliedsbeitrag als juristische Person (Institution) ❑

20.- DM für das Jahresabonnement der Mitteilungen ❑

auf das Konto **05 141 941 00** bei der Dresdner Bank Berlin BLZ **100 800 00**

❑ Ich erteile hiermit die **Einzugsermächtigung** für **_____,- DM** an die Deutsche Gesellschaft für Elektroakustische Musik e.V. von meinem Konto

Konto-Nr. .....

bei Bank..... BLZ.....

Name:..... Vorname:.....

Straße : .....

PLZ-Stadt : .....

Telefon : ..... / .....

FAX : ..... / .....

Email : .....

Datum: ..... **Unterschrift:** .....

Tobias Kunze
Rönnestr. 14, D–14057 Berlin
tkunze@gigant.kgw.tu-berlin.de

Common Music und Stella: Kompositionssprache und –umgebung in Lisp. Ein Überblick.

(Teil 2) [14]

3.2 Erzeugung Musikalischer Struktur

3.2.1 Erzeugung im Editor: `new`

Zur Erzeugung von Strukuren dient im Editor der Befehl `new`. Dabei ist es gleichgültig, welcher Objekttyp—welche Klasse—erzeugt werden soll, der Typ wird einfach als Argument angegeben.

```
Stella [Top-Level]: new midi-note 1 top-level
Slots and values: <cr>
Stella [Top-Level]:
```

oder häppchenweise

```
Stella [Top-Level]: new
Type of new object: midi-note
Number of midi-notes to create: (<cr>=*) 1
Slots and values: <cr>
New object position: (<cr>=Pasteboard) top-level
```

Beide Varianten erzeugen ein neues Objekt der Klasse *Midi-Note* im angegebenen Container *Top-Level*.

```
Stella [Top-Level]: list
Top-Level:
     1. #<MIDI -unset- -unset- -unset- -unset-  0 #xDC64F1>
```

Analog dazu erzeugt `new` einen neuen Thread mit

```
Stella [Top-Level]: new thread
Name for new Thread: tmp
New object position: (<cr>=Top-Level) <cr>
Stella [Top-Level]: list
Top-Level:
     1. #<MIDI -unset- -unset- -unset- -unset-  0 #xDC64F1>
     2. #<THREAD: Tmp>
Stella [Top-Level]:
```

Natürlich ist der Thread *Tmp* noch leer und enthält die so erzeugte MIDI–Note noch weder Tonhöhen– (*note*), noch Rhythmus– (*rhythm*), Dauern– (*duration*) oder Lautstärkeinformation (*amplitude*).

Um eine komplette Midi-Note mit allen nötigen Informationen zu erzeugen, hätte man auf die Frage nach Werten für die Slots (‘`Slots and values:`’) etwa angegeben:

```
[…]
Slots and values: note 'c4 rhythm 1 duration 4 amplitude .5
Stella [Top-Level]: list
Top-Level:
     1. #<MIDI   C4  1.000  4.000  0.500  0 #xF30371>
     2. #<THREAD: Tmp>
Stella [Top-Level]:
```

[14] Teil 1 wurde in den Mitteilungen_13 abgedruckt; siehe dort auch Liste der empfohlenen Literatur

Die Kanal-Information (der letzte Slot, '*channel*') hat als Voreinstellung '0'—also Kanal 1, wenn von 1 bis 16 gezählt wird—und muß deshalb nicht unbedingt angegeben werden.

Um Tmp mit (neuen) Elementen zu füllen, genügt es, auf die Frage nach ihrem Ort ('`New object position:`') einfach `tmp` einzugeben:

```
Stella [Top-Level]: new midi-note
Number of midi-notes to create: (<cr>=*) 3
Slots and values: note (notes c4 d e) rhythm (rhythms q e e) duration
$rhythm amplitude .5
New object position: (<cr>=Top-Level) tmp
Stella [Top-Level]: list tmp
Tmp:
     1. #<MIDI   C4  1.000  1.000  0.500  0 #xF4D601>
     2. #<MIDI   D4  0.500  0.500  0.500  0 #xF4D811>
     3. #<MIDI   E4  0.500  0.500  0.500  0 #xF4D8B1>
Stella [Top-Level]:
```

Bei dieser Initialisierung der Midi–Noten wird durch die Angabe `note (notes c4 d e)` bzw. `rhythm (rhythms q e e)` dem *note*– bzw. *rhythm*–Slot jedes Elementes der jeweils nächste Wert des *notes*– bzw. *rhythms*–Streams zugewiesen (zum Konzept dieser zusammenfassend mit *Item Streams* bezeichneten Datenstrukturen siehe unten unter 3.3).

3.2.2 Erzeugung über Lisp-Code

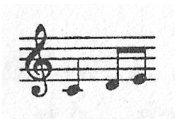
Obwohl Stella's interaktiver `new`–Befehl in vielen Fällen unproblematisch ist, stößt er doch—vor allem bei großen Mengen zu erzeugender Objekte oder kompexeren Berechnungen der Werte für die einzelnen Slots—an Grenzen. Hierfür, aber auch zur direkten Erzeugung zusammengesetzter Objekte eignen sich Lisp-Formen weitaus mehr. Daher bietet Stella für jede Container–Klasse einen gleichnamigen sog. *Constructor* sowie für Objekte einer Element-Klasse das Macro `object`. In eine iterative Form gebettet erzeugt `object` eine beliebige Anzahl von Elementen

```
? (loop for nte in '(c4 d4 e4)
    collect (object midi-note
                    note nte rhythm .1 duration .5
                    amplitude .5))
(#<MIDI   C4  0.100  0.500  0.500  0 #xF48ED1>
 #<MIDI   D4  0.100  0.500  0.500  0 #xF48F49>
 #<MIDI   E4  0.100  0.500  0.500  0 #xF48FC1>)
?
```

Solche Elemente können nun direkt als Elemente eines Containers erzeugt werden. Der Code

```
? (thread tmp ()
    (loop for nte in '(c4 d4 e4)
          for rhy in '(1 .5 .5)
          do (object midi-note
                     note nte rhythm rhy
                     duration .5 amplitude .5)))
#<THREAD: Tmp>
?
```

stellt denselben Thread *Tmp* mit den Noten



her wie die beiden `new`–Aufrufe oben (3.2.1). Die `thread`–Form erhält als Argumente einen Namen (*tmp*), eine (hier leere) Liste von Initialisierungen, sowie beliebige weitere

Formen, deren Resultate (Stella-Objekte) Bestandteile des Threads werden. Im Beispiel liegt nur eine solche —allerdings iterative— Form vor: `(loop …)`. Bei jedem Durchlauf der `loop`–Form werden die Variablen `nte` und `rhy` an das nächste Element ihrer Liste gebunden und mit diesen Werten für den *note*– bzw. *rhythm*–Slot dann `object` abgearbeitet.

Selbstverständlich können mittels einer solchen Schleife nicht nur Elemente sondern auch ganze Container rekursiv erzeugt werden. Dabei muß jedoch jedes erzeugte Exemplar einen eigenen Namen erhalten—ein aufgrund der Tatsache, daß die container–erstellende Form ein Makro ist, nicht ganz triviales Problem. Stella definiert daher eigens eine Funktion `name`

`name` *name* `&optional` *new*

die, wenn ihr Argument *new* nicht `nil` ist, mit jedem Aufruf eine neue Variante der als *name* angegebenen Namenswurzel liefert, etwa:

```
? (loop repeat 5 collect (name 'lemming- t))
(LEMMING-0 LEMMING-1 LEMMING-2 LEMMING-3 LEMMING-4)
?
```

3.2.2.1 Algorithms und Generators

Je mehr sich die Beschreibung einer kompositorischen (Teil–)Idee der Form nach einer Berechnung nähert, umso mehr empfiehlt sich natürlich die Verwendung von Lisp–Code. Die *algorithm*–Klasse und ihre Subklassen (*generator*) setzen sogar explizit Lisp-Code voraus. Ihre *Constructor*-Macros erwarten als Argumente einen Namen, den Namen der zu erzeugenden Element-Klasse, eine Liste bestehend aus beliebig vielen (null oder mehreren) Initialisierungen und als Rumpf ("*body*") der Form beliebig viele Formen, nämlich die algorithmische Beschreibung der zu erzeugenden Elemente—in Syntax–Notation also:

`algorithm` *name note-class* ({*inits*}*) `&body` *body*
`generator` *name note-class* ({*inits*}*) `&body` *body*

Der Algorithmus

```
(algorithm test midi-note (length 400 rhythm .125 duration 1 amplitude .5)
  (setf note (round (+ (* (sin count) 12) 72))))
```

generiert beispielsweise eine gleichförmige Abfolge von—entsprechend den Initialisierungen—400 (`length 400`) Zweiunddreißigstel-Noten (`rhythm .125`) im Mezzopiano (`amplitude .5`) und von einem Viertel Dauer (`duration 1`). Lediglich der *note*–Slot wird in Laufzeit berechnet. Allerdings erzeugt der Algorithmus nicht alle Elemente auf einmal: er alloziert vielmehr nur ein einziges Objekt der *midi–note*–Klasse, das nach Initialisierung seiner Slots wiederholt durch den Rumpf des Algorithmus umgeschrieben wird—und zwar so oft, wie seine Länge vorgibt. Durch diese Technik der Evaluierung, die Objekte nur erzeugt, soweit sie benötigt werden, wird wertvoller Speicherplatz und Verwaltungsaufwand eingespart. Die Berechnung des *note*–Slots greift auf einen internen *count*–Slot des Algorithmus zurück, der nach jedem Durchlauf des Rumpfes automatisch inkrementiert wird, also hier von 0 bis 399 hochgezählt wird. Somit ist die Tonhöhenfolge vollständig beschrieben als der gerundete Wert einer für die von `count` gegebenen ganzzahligen Radians–Werte von 0 bis 399 "abgetasteten" Sinuskurve, deren Wertebereich von –1 bis 1 im Verhältnis 1:12 skaliert sowie anschließend um 72 Halbtöne transponiert wurde, um im gewünschten Tonraum zwischen c4 (c') und c6 (c''') zu liegen (s. Abb. 3).

Zwar bringt der Algorithmus zahlreiche Scheinwiederholungen mit geringen Abweichungen hervor, ist aber natürlich als Prozeß insgesamt vollständig deterministisch—

d.h., jeder Wiederaufruf des Codes beispielsweise über Stella's `listen`–Befehl erzeugt zwangsläufig dieselbe Folge—und könnte somit ebensogut (und jederzeit) als Generator (re–)definiert werden. In der Regel hängt jedoch die Entscheidung, eine (deterministische) Berechnung als Generator oder als Algorithmus zu implementieren von anderen Präferenzen ab, etwa ob in der gegebenen Situation Einsparungen in der Speicherbelegung effizienteren Rechenprozessen vorzuziehen seien oder nicht, usf.

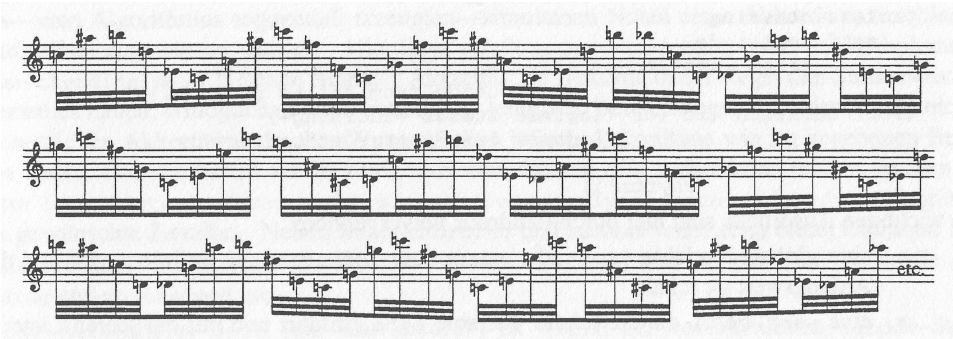


Abb. 3: Algorithm 'Sinus'

3.2.2.2 Komplexere Konstrukte in Algorithmen

Komplexere Algorithmen verwalten selbstverständlich in der Regel mehr als einen Slot, greifen häufig auf lokale oder sogar globale Variablen und Datenstrukturen zurück und gründen oft Teile der Berechnung auf aktuelle Werte eines Slots oder Beziehungen zwischen aktuellen Werten mehrerer Slots. Die für solche Programmkonstrukte nötigen Kontrollstrukturen stammen jedoch nicht alle aus dem Vokabular von Common Lisp. Da die *Constructor*–Macros der Algorithm– wie Generator–Klasse nicht zuletzt aufgrund der diesen Objektklassen inhärenten iterativen Form in wesentlich komplizierteren Code expandieren als ihr Aufruf vermuten läßt, können manche Konstrukte nicht leicht aus normalen Lisp–Formen aufgebaut werden. Daher stellt Stella spezielle Formen dieser Konstrukte eigens zur Verwendung innerhalb von Algorithmen bereit. Beispielsweise erlaubt die `vars`–Form, am Beginn eines Algorithmenrumpfs lokale Variablen *für die Dauer des gesamten* Algorithmus zu vereinbaren—eine `let`–Form an dieser Stelle würde dagegen *mit jedem Durchlauf* das Algorithmenrumpfes abgearbeitet werden.

Das folgende Codebeispiel versammelt um der Vollständigkeit willen alle Typen der genannten Konstrukte auf—zugegebenermaßen—engstem Raum:

```
(defparameter *top* 84)

(algorithm solo midi-note ()
  (vars (echo ()))
  (setf note (item (items ((intervals 0 -1
                                      for (items 1 2 in random)
                                      from (expr *top*)
                                      named trill)
                           weight 6 max 2)
                          (items in function
                                 with #'(lambda ()
                                          (if echo
                                            (prog1
                                              (nreverse echo)
                                              (setf echo ()))
                                            cm::+nad+)))
                          (chord (intervals -41 -34 -26 -18 -9  9
                                            from #@trill))
                          r
```

```
                          in random for 9)
                   :kill 3))
  (when (<= 60 note 82) (push note echo))
  (when (>= note 84) (setf *top* note))
  (setf rhythm (if (status? :resting) e. s))
  (unless-chording
    (setf amplitude
          (item (amplitudes (crescendo from pp to f in 8)
                            (diminuendo from fff to pp in 5)))))
  (setf duration (if (or (logtest status +chording+)
                         (logtest last-status +chording+))
                   h.
                   rhythm)))
```

Die wichtigen Ausdrücke sind hier durch Fettdruck hervorgehoben:

- eine globale Variable `*top*`, die extern über `defparameter` definiert wird und den Anfangswert `84` erhält,
- eine `vars`–Form, die die lokale Variable `echo` einführt und mit der leeren Liste `()` initialisiert,
- die spezielle, nur innerhalb von Algorithmen zulässige Form der Funktion `item`: `(item … :kill` *n*`)`, die den Algorithmus nach drei Durchläufen des eingeschlossenen Item Streams (s. 3.3) `(items … in random for 9)` abbricht (weshalb hier keine Initialsierung des `lenght`–Slots wie im ersten Beispiel oben nötig ist),
- die "Veröffentlichung" eines Item Streams mittels der Option `named` und dessen namentlicher Referenzierung über `#@`*name*, sowie
- verschiedene Kontrollstrukturen, die direkt oder indirekt mit den Werten der Slots `status` und `last–status` umgehen, vor allem die Funktionen `status?` und `unless-chording`.

Den Kern des Algorithmus bildet die Erzeugung des musikalischen Rohmaterials durch den Item Stream `(items … in random for 9)`. Er liefert in zufälliger Reihenfolge neun Produkte seiner vier Elemente und wird durch (item … :kill 3) nach drei Perioden, also insgesamt 18 hervorgebrachten Mustern abgebrochen. Die erste Subform `((intervals …) weigth 6 max 2)` ist dabei sechsfach gewichtet mit der Maßgabe, nicht öfter als zweimal direkt hintereinander ausgewählt zu werden und enthält den `intervals`–Stream namens `trill`, der pro Aufruf ein oder zwei Zahlen im kontinuierlich alternierenden Intervallabstand 0 und -1 vom gerade aktuellen Wert der Globalen `*top*` zurückliefert. Die zweite Subform `(items in function with` *function*`)` liefert durch ihren Funktionsrumpf

```
(if echo
  (prog1
    (nreverse echo)
    (setf echo ()))
  cm::+nad+)
```

etwaige in der lokalen Variable `echo` angesammelten Noten und reinitialisiert `echo` anschließend mit der leeren Liste. Die dritte Subform `(chord (intervals … from #@trill))` faßt sechs ausgehend vom gegenwärtigen Wert des Item Streams `trill` erzeugten Intervalle zu einem Akkord zusammen und die vierte schließlich, `r`, ist eine schlichte Pause.

Mit jedem Durchlauf des Algorithms nun wird der `note`–Slot auf den nächsten Wert dieses in Form eines Item Streams vorliegenden musikalischen Rohmaterials gesetzt und—von ihm ausgehend—durch die beiden folgenden `when`–Formen etwaige Veränderungen an `echo` und `*top*` vorgenommen: fällt eine Note zwischen c4 und b5, wird sie auf den `echo`–Stack geschoben; ist sie aber höher als c6, wird sie zur neuen `*top*`–Note und beeinflußt so die

weiteren Berechnungen der erwähnten beiden `intervals`–Streams. Dann erfolgt die Berechnung der übrigen Slots aufgrund des jeweiligen Algoritmus–Status. Durch `(if (status? :resting) e. s)` beispielsweise wird sichergestellt, dass alle Noten den rythmischen Wert eines Sechzehntels, Pausen aber die eines punktierten Achtels erhalten, und die folgende Form `(unless–chording …)` verhindert, dass den — vom Algorithmus sequentiell erzeugten — simultanen Noten eines Akkords unterschiedliche Lautstärken zugewiesen werden. Allerdings erfaßt `unless-chording` (oder das komplementäre `when–chording`) *nicht* die erste Note des Akkordes: da Akkorde in der Regel den Stellenwert *eines* Elementes haben, ermöglichen diese Funktionen lediglich, Berechnungen nur oder gerade nicht für die restlichen Akkordtöne durchzuführen. Sollen jedoch Akkordtöne wie im gegebenen Beispiel eine andere Dauer erhalten als die übrigen Noten, müssen *alle* Akkordtöne isoliert werden. Die Form `(or (logtest status +chording+) (logtest last–status +chording+))` gibt hier das gewünschte Resultat. Neben `when–chording` und `unless–chording` bietet Common Music noch eine Reihe gleichartiger Funktionen wie `when–resting` und `unless–resting` sowie `when–ending` und `unless–ending`.

Der Algorithmus `solo` (ein Klaviersolo?) ist wegen der verschiedenen `random`–Streams natürlich nicht deterministisch. Aber auch die externe Variable `*top*` wird nicht mit jedem Mal neu auf den — arbiträren — Initialwert 84 gesetzt, weshalb er bei mehrmaligem Aufruf bisweilen *äußerst* unterschiedliche Resultate bringen kann. Abb. 4 stellt aber eine hinlänglich repräsentative Version dar.

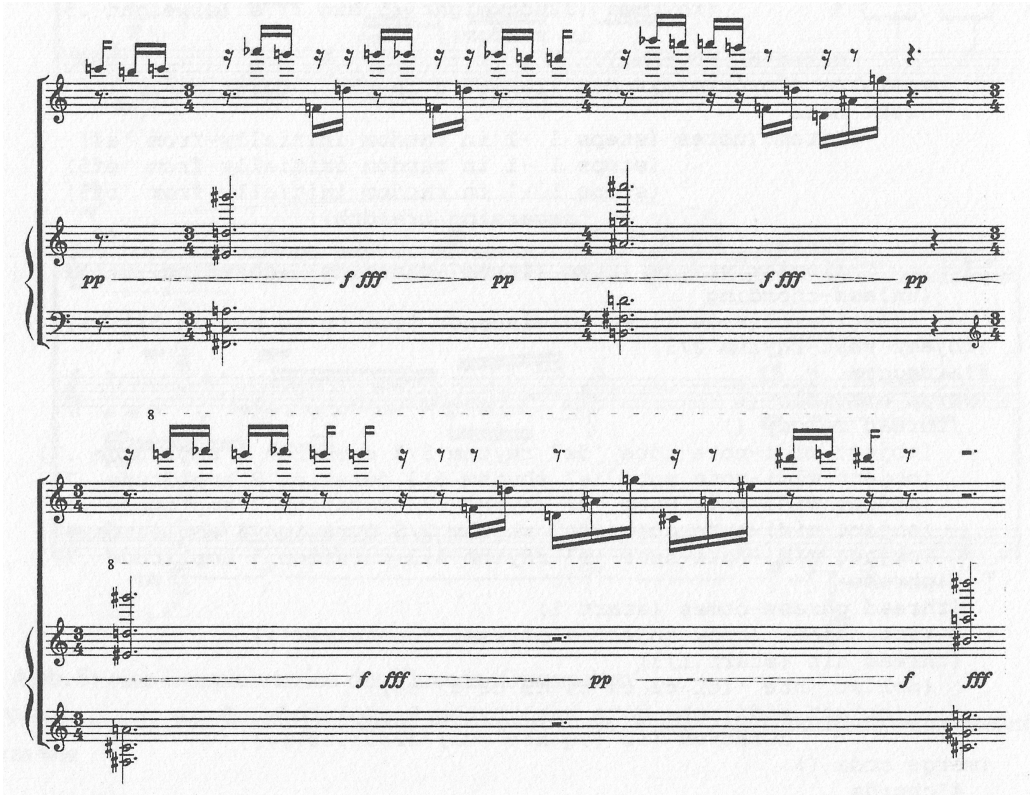


Abb. 4: Eine Version des Algorithms ‘Solo’.

3.2.3 Musikalische Struktur und Layout

Die musikalische Struktur einer Komposition repräsentiert Stella zwanglos in Form einer Hierarchie von Elementen und Containern: einzelne oder mehrere zu einem Container zusammengefasste Elemente bilden zugrundeliegende Einheiten und sind typischerweise in möglicherweise verschiedenen übergeordenten Containern enthalten, die je nach Klassenzugehörigkeit—Merge, Thread, Heap, usf.—ihren zeitlichen und grob formalen Bezug zueinander regeln. Solche Strukturkomponenten können dabei beliebig tief und an beliebig vielen Stellen ineinander geschachtelt werden.

Das folgende Programmstück zeigt, wie mit relativ einfachen Mitteln etwa die Rohfassung einer kleinen Komposition modelliert werden könnte (eine durchschnittliche Version der durch diesen Code erzeugten Musik gibt Abb. 5 wieder):

```
(thread opusculum ()
  (object rest rhythm 2/3)
  (object midi-note note 'b1 rhythm 0 duration 22/3 amplitude .5)
  (heap arpeggio ()
    (dolist (nte '(fs2 a3 f4 gs4 e5 g5 ef6))
      (object midi-note
              note nte rhythm 1/3 duration 1/3 amplitude .4)))
  (merge theme ()
    (algorithm phrase midi-note (length 12 note 60 amplitude .2)
      (setf note (+ note (- (random 5) 2))
            rhythm (item
                     (rhythms (32nd weight .5 min 2) s (e weight .5)
                              in random))
            duration rhythm))
    (generator chords midi-note (length 5 rhythm 1 duration 1)
      (setf note
            (item (notes (steps 1 -1 in random initially-from 'a4)
                         (steps 1 -1 in random initially-from 'ef5)
                         (steps 1 -1 in random initially-from 'bf5)
                              traversing breadth))
            status
            (logior status (item (items +chording+ +chording+ 0))))
      (unless-chording
        (setf amplitude (item (diminuendo from pp to ppp in 5))))))
  (object rest rhythm 2/3)
  #!arpeggio
  (merge ensemble ()
    (thread melody ()
      (object midi-note note 'ds7 rhythm 5/3 duration 3 amplitude .7)
      (object midi-note note 'e7 rhythm 6/3 duration 3 amplitude .7)
      (object midi-note note 'd7 rhythm 1/3 duration 3 amplitude .7)
      (object midi-note note 'fs7 rhythm 2/3 duration 3 amplitude .7)
      (object midi-note note 'f7 rhythm 4/3 duration 3 amplitude .7))
    #!phrase
    (thread phrase-comes (start 1)
      (add-object (copy-object #!phrase) *container*))
    (thread hit (start 1/3)
      (dolist (nte '(c1 c2 c3 c4 c5 c6 c7 c8))
        (object midi-note note nte rhythm 0 amplitude .8
                duration (if (eq nte 'c2) 22/3 .05)))))
  (merge coda ()
    #!chords
    (object midi-note start 5/3
            note 'bf1 rhythm 0 duration 22/3 amplitude .5)))
```

Der grobe musikalische Ablauf ist durch den äußersten `(thread opusculum () … )` vorgegeben: das Triolen–Viertel Pause (`(object rest rhythm 2/3)`) und die beiden darauf

zeitgleich (`rhythm 0`) folgenden Objekte—ein lang gehaltener Baßton `b1` sowie ein aus sieben weiteren Triolen–Achteln bestehender `(heap arpeggio () … )`—bilden einen ersten Abschnitt. Daran schließt ein 'fester' gefügter, aus unten liegendem rythmisiertem Stufengang (`(algorithm phrase … )`) und metrisch gleichförmig bewegter Akkorbegleitung (`(generator chords … )`) bestehender `(merge theme … )` an. Nach einer Wiederholung der drei Viertel dauernden Kombination aus Pause und Triolenarpeggio—die Notation '`#!`<*name*>' ist gleichbedeutend mit `(find–object` <*name*>`)`—setzt dann ein durch– und engführender `(merge ensemble … )` ein: über zwei um ein Viertel versetzten Varianten der `phrase` genannten Tonbewegung (`#!phrase` und `(thread phrase-comes (start 1) … )`) sowie einem kurzen *sforzato*–Akkord aller `c`–Oktaven auf dem zweiten Triolenachtel (`(thread hit (start 1/3) … )`)—mit dem zusammen auch wieder der lang gehaltene Baßton, diesmal auf `c2` einsetzt: `duration (if (eq nte 'c2) 22/3 .05)`—liegt in der obersten Oktave des Klavierumfangs eine relativ langsame, in Triolen zählende melodische Linie (`(thread melody () … )`). Den Abschluß bildet der `(merge coda () … )` mit einer wörtlichen Wiederaufnahme des (nunmehr 'eingefrorenen') Generators `chords` und einer rythmisch versetzten Rückführung des noch klingenden Basses `c2` auf seine Ausgangstonhöhe `b1` und –lautstärke `.5` (`(object midi-note start 5/3 … )`).

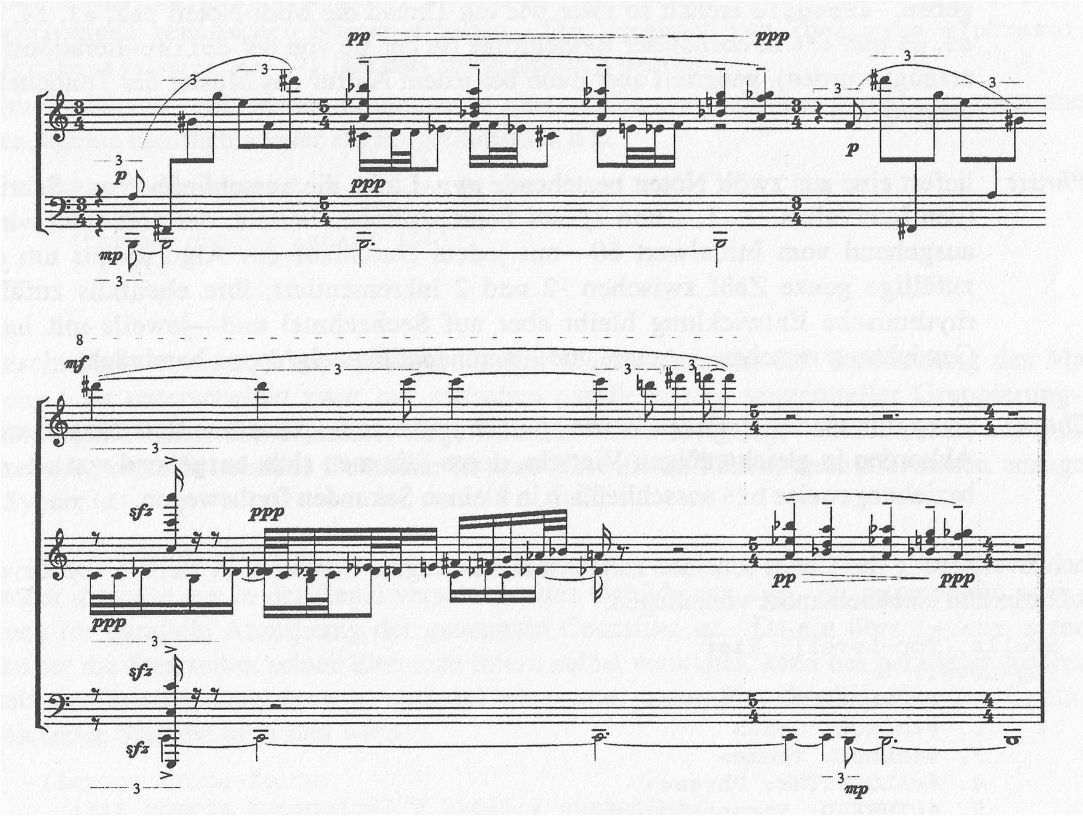


Abb. 5: eine Beispielversion des Threads 'Opusculum'

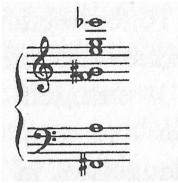
Wird `chords` vor jedem Durchlauf von `opusculum` reinitialisiert, beispielsweise mittels einer Form wie

```
(progn
  (unfreeze-object #!chords)
  (listen-object #!opusculum))
```

dann enthält die musikalische Gesamtgestalt des Threads `opusculum` durch die Container `arpeggio`, `phrase` und `chords` an insgesamt sieben Stellen bewegliche Komponenten—eine so etwa zu Entwurfszwecken angelegte Rohskizze kann dann in schneller Folge beliebig

viele Varianten des gewünschten Ablaufes erzeugen, wobei sich die musikalische Variationsbreite in jedem einzelnen Fall genau angeben läßt:

Arpeggio Resultat des Heaps `arpeggio` ist immer eine Auflösung des Akkordes



in eine Kette von sieben ***mp***–Triolen–Achteln: die Klasse *Heap* ist, wie aus Abb. 2 hervorgeht, eine Subklasse von Thread, die den ererbten Slots keine neuen hinzufügt, sondern sich ausschließlich in ihrem *Verhalten* von Threads unterscheidet—die Elemente eines Heaps werden jedes Mal in veränderter, zufälliger Reihenfolge ausgegeben. `arpeggio` enthält so zwar wie ein Thread die Midi-Noten `fs2`, `a3`, `f4`, `gs4`, `e5`, `g5` und `ef6` in ebendieser Reihenfolge (in der sie von der `dolist`–Iterationsform erzeugt wurden), generiert aber dann bei jedem Aufruf das Muster der Tonhöhenbewegung neu.

Phrase liefert eine aus zwölf Noten bestehende ***ppp***–Linie, die ausschließlich aus Schritten (den Intervallen -2, -1, 1 und 2) und Tonrepetitionen besteht: der *note*–Slot wird—ausgehend vom Initialwert 60—mit jedem Durchlauf des Algorithmus um eine zufällige ganze Zahl zwischen -2 und 2 inkrementiert; ihre ebenfalls zufällige rhythmische Entwicklung bleibt aber auf Sechzehntel und—jeweils mit halber Gewichtung versehen— Achtel und Zweiunddreißigstelgruppen beschränkt.

Chords erzeugt in einem *diminuendo* von ***pp*** nach ***ppp*** eine Sequenz von fünf dreistimmigen Akkorden in gleichmäßigen Vierteln, deren Stimmen sich, ausgehend von `a4`, `ef5`, beziehungsweise `bf5` ausschließlich in kleinen Sekunden fortbewegen.

Nach Evaluierung des Codes scheinen zunächst alle erzeugten Container in Stella's *Top-Level* in unstrukturiertem Nebeneinander vorzuliegen:

```
Stella [Top-Level]: list
Top-Level:
     1. #<GENERATOR: Chords>
     2. #<MERGE: Coda>
     3. #<MERGE: Theme>
     4. #<ALGORITHM: Phrase>
     5. #<THREAD: Phrase-Comes>
     6. #<MERGE: Ensemble>
     7. #<ALGORITHM: Algorithm-17>
     8. #<THREAD: Melody>
     9. #<THREAD: Hit>
    10. #<THREAD: Opusculum>
    11. #<HEAP: Arpeggio>
Stella [Top-Level]:
```

Die genaue Gestalt der musikalischen Struktur zeigt sich dann bei der Anwendung des `list`–Befehls auf die einzelnen Container.

```
Stella [Top-Level]: list opusculum
Opusculum:
     1. #<REST   0.667 #xF41939>
     2. #<MIDI   B1  0.000  7.333  0.500  0 #xF419A1>
     3. #<HEAP: Arpeggio>
     4. #<MERGE: Theme>
     5. #<REST   0.667 #xF42561>
     6. #<HEAP: Arpeggio>
     7. #<MERGE: Ensemble>
     8. #<MERGE: Coda>
Stella [Top-Level]: list ensemble
Ensemble:
     1. #<THREAD: Melody>
     2. #<ALGORITHM: Phrase>
     3. #<THREAD: Phrase-Comes>
     4. #<THREAD: Hit>
Stella [Top-Level]:
(…etc…)
```

Der in Top-level an siebenter Stelle stehende Algorithm "Algorithm–17" ist das Resultat des Aufrufs `(copy-object #!phrase)` im Rumpf von `phrase-comes` und hat auf diesem Weg lediglich einen generischen Namen erhalten. Ein Vergleich von `(describe #!phrase)` und `(describe #!algorithm-17)` läßt dies sofort deutlich werden.

Um die bereits vorhandenen Container zu alternativen musikalischen Strukturen zusammenzusetzen, könnte natürlich wieder ein Programmstück à la

```
(merge new-opusculum ()
  #!theme
  (thread chorus (start 100)
    #!hit
    ... )
  ... )
```

geschrieben werden. Viel einfacher ist hier jedoch meistens die Verwendung des Makros `layout`. Es unterscheidet zwar nur zwischen paralleler und sequentieller Gruppierung—die Klassenzugehörigkeit der Container ist ja ohnehin schon festgelegt—erlaubt aber, auf einfache Weise, d.h., ohne bestehende Container in neue Threads zu hüllen, neue Startzeiten anzugeben. Die Syntax ist

`layout` *name directive*

wobei *directive* ein in der Regel verschachteltes Gebilde aus `seq`– für sequentielle oder `mix`–Formen für parallele Anordnung der genannten Container ist. Da ein über `layout` erzeugter Container die Startzeiten seiner Elemente intern selbst verwaltet, kann bei paralleler Anordnung mittels des Postfix–Operators `@` optional eine vom *start*–Slot des betreffenden Containers abweichende Startzeit bestimmt werden.

```
(layout brute–force
  (mix chords arpeggio@2/3 hit@1.2 ensemble@2
       (seq hit hit hit hit)@2.9))
```

erzeugt einen Merge *Brute–Force* mit den angegebenen Elementen und Startzeiten sowie einen neuen Thread mit generischem Namen für die Elemente der `seq`–Form:

```
Stella [Top-Level]: list brute–force
Brute–Force:
     1. #<GENERATOR: Chords>
     2. #<HEAP: Arpeggio>
     3. #<THREAD: Hit>
     4. #<MERGE: Ensemble>
     5. #<THREAD: Thread-17>
Stella [Top-Level]: list thread-17
```

```
Thread-17:
     1. #<THREAD: Hit>
     2. #<THREAD: Hit>
     3. #<THREAD: Hit>
     4. #<THREAD: Hit>
Stella [Top-Level]:
```

3.2.4 Dynamische Strukturdefinition: `sprout` und `mute`

Erlaubt eine Kompositionsidee nicht, Zugehörigkeiten oder Startzeiten ihrer Container oder sogar die Anzahl oder Art der benötigten Container im Voraus zu bestimmen, so muß die musikalische Struktur teilweise oder auch zur Gänze während der Laufzeit des Codes erstellt werden. Da die einer solchen dynamischen Strukturdefinition zugrundeliegenden Berechnungen ebenso zeitlich kontrolliert ausgeführt werden müssen wie bei der direkten Erzeugung innerhalb eines Algorithmus, definiert Stella eigens eine spezielle Art von "stummen" Algorithmen, d.h., Algorithmen mit der Element–Klasse `nil` (anstelle von etwa `midi-note`). Innerhalb dieser *Mute* genannten, stummen Algorithmen können an beliebiger Stelle dynamisch neue Objekte erzeugt werden. Die Funktion `sprout` erledigt dabei die im Detail komplizierte korrekte Einbindung des—möglicherweise andere Objekte enthaltenden—Objektes in den gerade ausführenden Prozess.

Das folgende Beispiel überlagert eine zugrundeliegende Gestalt mit zwei Variationen ihrer durch Akzentuierungen hervorgehobenen Hauptlinie (des 'Themas') in zwei– bzw. vierfacher Augmentation (s. dazu auch Abb. 6). Die für den kontrollierten Ablauf des musikalischen Geschehens zentralen Formen sind dabei durch Fettdruck hervorgehoben:

```
(mute ghosts (length 12)
  (let (tim nte main? rhy amp)
    (setf tim (* (+ time .5) 2)
          nte (between 53 77)
          main? (>= nte 65)
          rhythm (item (rhythms s e e. in random))
          rhy rhythm
          amp (if main? .6 .4))
    (sprout
     (object midi-note start time
             note nte rhythm 0 duration rhy amplitude amp))
    (if main?
      (sprout
       (thread nil (start tim)
         (object midi-note
                 note (+ nte 24) rhythm tim duration tim
                 amplitude .5)
         (algorithm nil midi-note (rhythm (/ rhy 4) amplitude .3
                                   duration 10)
           (setf note
                 (item (items in function with
                               #'(lambda ()
                                   (loop for n from (+ 39
                                                       (mod nte 13))
                                                    by 13
                                         repeat 5
                                         collect n)))
                       :kill t)))))
      (when (= rhy .75)
        (sprout
         (thread nil (start (+ time .5))
           (dolist (n (list (- nte 18) (- nte 23)))
             (object midi-note
                     note n rhythm 0 duration .05
                     amplitude .4)))))))))
```

Die erste `setf`–Form weist mit jedem Durchlauf den durch die äußerste `let`–Form eingeführten Laufzeitvariablen einen neuen Wert zu. Im Einzelnen erhält `tim` den jeweils doppelten Wert der gerade aktuellen Laufzeit plus 0.5 (d.h., der dem 'Thema' und seinen zwei Variationen gemeinsame virtuelle Einsatzzeitpunkt liegt ein Achtel vor Beginn des Mutes, vgl. das Notenbeispiel Abb. ???), `nte` eine zufällig gewählte Note zwischen `53` (*f3*) und `77` (*f5*), `rhy` den Wert des *rhythm*–Slots, entweder ein Sechzehntel, ein Achtel oder ein punktiertes Achtel, sowie `amp`—abhängig von `main?`—entweder `.4` (***p***) oder `.6` (***mf***, hier aber sozusagen ***p*** mit Akzent). `main?` selbst ist eine Flag und hat den Wert `t`, wenn `nte` größer als `65` (*f3*) ist. Diese Laufzeitvariablen bilden zusammen die Basis der im Folgenden ausschließlich deterministisch verlaufenden Berechnung.

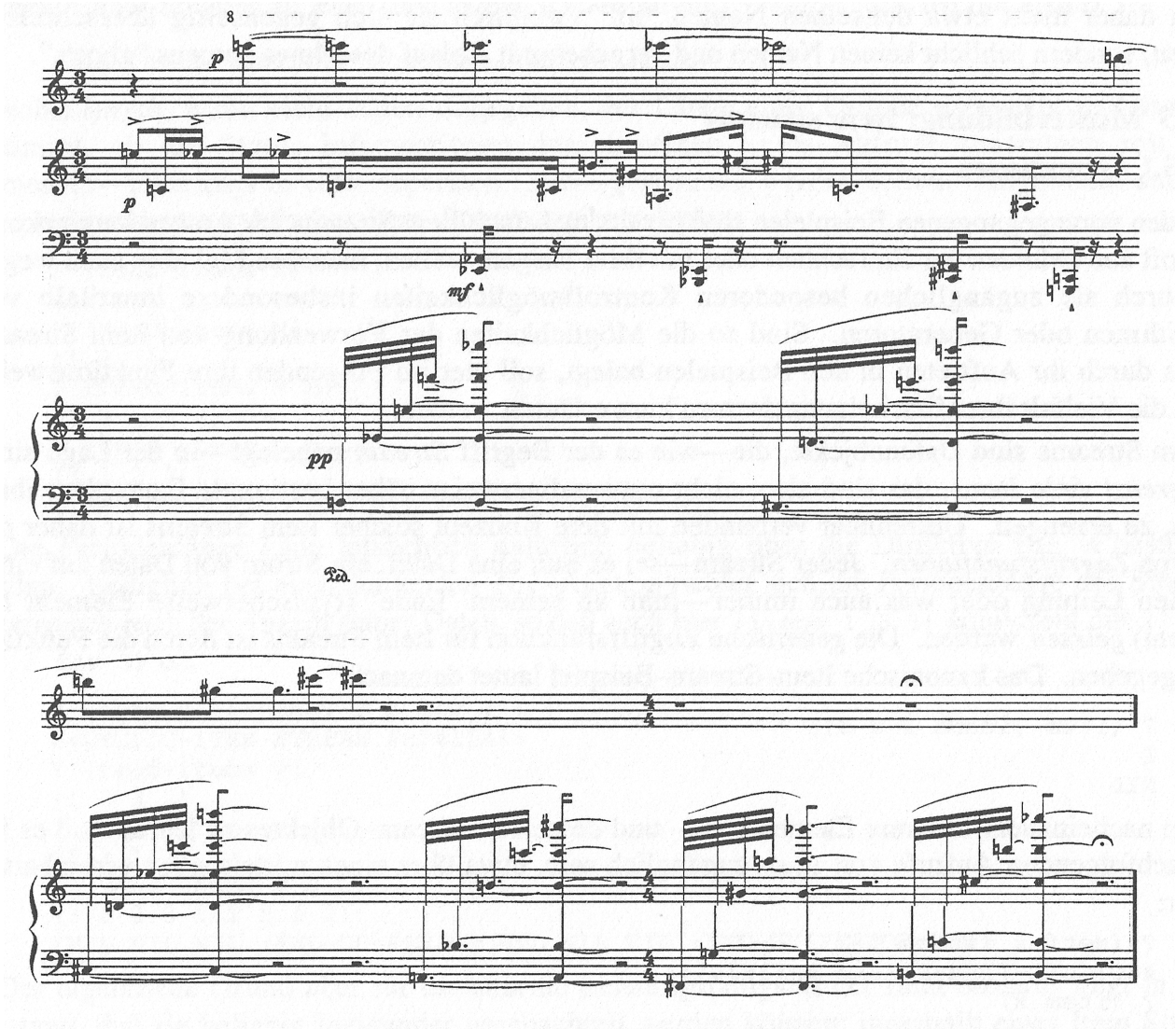


Abb. 6: eine Beispielversion des Mutes 'Ghosts'

Zunächst wird eine `sprout`–Form aufgerufen, um eine aus diesen Variablen zusammengesetzte Midi-Note zu erzeugen—die Originalversion also auch wirklich hörbar werden zu lassen. Der *rhythm*–Slot der erzeugten Note—der zeitliche Abstand zum nächsten Ereignis—ist dabei, da sie als Einzelereignis auftritt, ohne Bedeutung. Allerdings benötigt sie als ein solches Einzelereignis eine eigene Zeitinformation, weshalb ihr `start`–Slot mit der aktuellen Zeit (`time`) des Mute als Zeitstempel versehen wird.

Die nächste Form entscheidet aufgrund der Flag `main?`, ob die erzeugte Note von den Augmentationen variiert wird: hat `main?` den Wert `t`, d.h., liegt der aktuelle Wert von `nte`

über 64, so wird zum Startzeitpunkt `tim` eine um zwei Oktaven nach oben versetzte ***mp***–Note sowie, noch einmal *tim* Viertel später, ein um die Tonhöhe `nte` herum gebauter fünftöniger Akkord aus übereinandergeschichteten kleinen Nonen ***pp*** und pedalisiert (`duration 10`) arpeggiert. Das Tempo des Arpeggios ist dabei stets ein Viertel der Dauer (`rhy`) der Originalnote.

Hat `main?` dagegen den Wert `nil`, so wird lediglich für den Fall, daß die Dauer der Originalnote gerade ein punktiertes Achtel beträgt im zeitlichen Abstand eines Achtels ein ***p***–Quart–Zweiklang eine Oktave und einen Tritonus unter der Tonhöhe der Originalnote *staccatissimo* angeschlagen.

Die so dynamisch erzeugten Objekte machen natürlich in der Regel wenig Sinn und sollten deshalb auch nicht als Container in Stella's Top-Level erscheinen. Erhalten solche Container einfach `nil` anstelle eines Namens (der dem System als Identifikator dienen würde), endet ihre Existenz mit dem Ende der Laufzeit des übergeordeneten Mutes. Die beiden Threads im Beispiel haben daher nicht etwa denselben Namen "nil" (wodurch sie sich gegenseitig überschreiben würden) sondern schlicht keinen Namen und vergehen mit Ablauf des Mutes namens "ghosts".

3.3 Musterbildung: Item streams

3.3.1 Grundlagen

In den vorangegangenen Beispielen sind bereits mehrfach Item Streams zur Anwendung gekommen, oft aus Gründen der Einfachheit oder um ihrer Eleganz willen, manches Mal aber auch wegen der durch sie zugänglichen besonderen Kontrollmöglichkeiten insbesondere innerhalb von Algorithmen oder Generatoren. Sind so die Möglichkeiten der Verwendung von Item Streams bereits durch ihr Auftreten in den Beispielen belegt, soll hier im Folgenden ihre Funktionsweise sowie die Vielfalt ihrer Erscheinungsformen kurz erläutert werden.

Item Streams sind Datenobjekte, die—wie es der Begriff *Stream* nahelegt—in der Lage sind, *unbegrenzt* viele *Items*, das sind also: nicht notwendigerweise näher bestimmte Exemplare ihrer Daten, zu erzeugen. Untrennbar verbunden mit dem Konzept solcher Item Streams ist daher die Idee von *Zugriffsfunktionen*. Jeder Stream—sei es nun eine Datei, ein Strom von Daten auf einer seriellen Leitung oder was auch immer—muß an seinem 'Ende' (typischerweise Element für Element) *gelesen* werden. Die generische Zugriffsfunktion für Item Streams ist durch die Funktion `item` gegeben. Das kanonische Item–Stream–Beispiel lautet demnach:

```
? (item (items 1 2 3))
1
NIL
```

Um nacheinander mehrere Elemente ein– und desselben Stream–Objektes zu lesen, muß es für die nachfolgenden Aufrufe von `item` zugänglich sein, etwa über einen mittels `setf` vereinbarten Zeiger:

```
? (setf x (items 1 2 3))
#<CYCLIC-ITEM-STREAM #xF3E7C1>
? (item x)
1
NIL
? (item x)
2
NIL
? (item x)
3
:END-OF-PERIOD
```

```
? (item x)
1
NIL
? (item x)
2
NIL
... etc. ...
```

Der jeweils zweite von `item` zurückgelieferte Wert—`nil` bzw. `:end–of–period` im letzten Beispiel—dient dazu, zum Element gehörige Statusinformation auszugeben. Dabei ist hier allerdings nur die Information `:end–of–period` von Bedeutung. Der unendliche Stream von Elementen ist nämlich aufgebaut aus von ebendiesen Elementen—seinen Daten—errichteten *Perioden*. Die Art und Weise, wie diese Perioden aus den vorhandenen Daten zusammengesetzt werden ist dabei abhängig vom *Pattern* des Item Streams. Das Beispiel oben zeigt, daß Item Streams, sofern kein anderes Pattern—etwa `in random`—angegeben wird, eine unendliche *zyklische* Folge ihrer Daten generieren. `(items 1 2 3)` ist somit gleichbedeutend mit `(items 1 2 3 in cycle)`.

Zusätzlich zur Grundfunktion `item` bietet `read–items` die Möglichkeit, auf einmal mehrere Elemente eines Streams zu lesen (und zugleich zu sammeln). `read–items` hat die Syntax

`read–items` <*stream*> `&optional` *length states?*

wobei *length*, wenn nicht anders angegeben, der Länge einer Periode entspricht und *status?* bestimmt, ob die Status–Informationen der einzelnen `item`–Aufrufe zusammen mit den Elementen—allerdings in einer separaten Liste—gesammelt werden sollen. Angewandt auf den von `x` referenzierten zyklischen Item–Stream ergibt sich also:

```
? (read-items x)
(3)
? (read-items x)
(1 2 3)
? (read-items x 5)
(1 2 3 1 2)
? (read-items x 5 t)
(3 1 2 3 1)
(:END-OF-PERIOD NIL NIL :END-OF-PERIOD NIL)
```

Die Periodenlänge kann tatsächlich natürlich beliebig über die Direktive `for` *n* bestimmt werden. Lediglich bei einer fehlenden `for`–Angabe richtet sich die Periodenlänge des Streams automatisch nach der Anzahl seiner Daten, so daß auch hier `(items 1 2 3)` äquivalent mit `(items 1 2 3 for 3)` ist.

```
? (setf y (items 1 2 3 for 4))
#<CYCLIC-ITEM-STREAM #xF41541>
? (read-items y)
(1 2 3 1)
? (read-items y)
(2 3 1 2)
? (read-items y 9 t)
(3 1 2 3 1 2 3 1 2)
(NIL NIL NIL :END-OF-PERIOD NIL NIL NIL :END-OF-PERIOD NIL)
```

Der eigentliche Grund aber für die enorme Leistungsfähigkeit der Item Streams liegt in dem Umstand, daß sie beliebig ineinander geschachtelt werden können: innerhalb eines Item Stream auftretende weitere Item Streams werden bei Errichtung des Streams rekursiv expandiert und gehen so in das vom äußersten Stream erzeugte Gesamtmuster ein. Dabei ist es gleichgültig, ob eingeschlossene Item Streams anstelle eines Elementes oder möglicherweise als Wert einer Item–Stream–Option—etwa `for` *n*— erscheinen! Bereits simple Programmstücke sind so in der Lage, relativ komplexe Abläufe zu erzeugen:

```
? (read-items (items 1 2 (items a b c d for (items 2 0 3 1))) 50)
(1 2 A B 1 2 1 2 C D A 1 2 B 1 2 C D 1 2 1 2 A B C 1 2 D 1 2 A B 1 2 1 2 C
D A 1 2 B 1 2 C D 1 2 1 2)
```

Die Analyse des Beispiels zeigt, wie der äußere Stream seinem zyklischen Pattern–Type entsprechend immer wieder reihum `1`, `2` sowie *eine volle Periode* des an dritter Stelle stehenden Substreams ausgibt. Der innere Stream jedoch hat keine konstante Periodenlänge und gibt so —ebenfalls zyklisch— 2, 0, 3 bzw. 1 Element seiner Daten `a`, `b`, `c` und `d` aus. Da dieser Substream selbst wieder zyklisch ist, finden sich seine Elemente in der Gesamtsequenz schließlich nacheinander als {`a b`}, {}, {`c d a`}, {`b`}, {`c d`}, usf. wieder. Dieses Verhalten des äußeren Streams—gewissermaßen zunächst jedem auch noch so tief geschachtelten Substream nachzugehen und sein Periodenende abzuwarten, bevor zum nächsten Element auf nächsthöherer Ebene übergegangen wird—mag jedoch nicht immer gewünscht sein. Mittels der Option `traversing` <*value*> kann dieses Standardverhalten umgangen werden:

```
? (read-items (items 1 2 (items a b c d for (items 2 0 3 1))
                     traversing breadth) 50)
(1 2 A 1 2 B 1 2 1 2 C 1 2 D 1 2 A 1 2 B 1 2 C 1 2 D 1 2 1 2 A 1 2 B 1 2 C
1 2 D 1 2 A 1 2 B 1 2 1 2)
```

`breadth` steht hier für ‘breadth–first’ entgegen dem genannten normalen `depth`–(‘depth–first‘)–Vorgehen. Zwar werden die Elemente des Substreams immer noch mit verschiedenen Periodenlängen erzeugt, aber nur noch jeweils ein Element geht in den laufenden Prozess der Mustererzeugung ein. Ebenso sind zwar die tatsächlichen Periodengrenzen des Substreams genauso nachvollziehbar wie im vorigen Beispiel, aber wegen seines zyklischen Pattern–Types von außen (d.h., ohne Kenntnis des Codes) nur noch im Falle der Null–Periodenlänge direkt ablesbar.

Weiterhin kann es manchmal entscheidend sein, eine *exakte* Anzahl von Elementen pro Periode zu erhalten. In solchen Fällen erlaubt die Option `counting` <value>, die Voreinstellung `periods` durch `values` aufzuheben:

```
? (setf z (items a (items 1 2 3) b (items 4 5) c for (items 2 3 1 4)))
#<CYCLIC-ITEM-STREAM #xF4A409>
? (read-items z)
(A 1 2 3)
? (read-items z)
(B 4 5 C)
? (read-items z)
(A)
? (read-items z)
(1 2 3 B 4 5 C)
? (read-items z)
(A 1 2 3)
... etc. ...
? (setf z (items a (items 1 2 3) b (items 4 5) c
                for (items 2 3 1 4) counting values))
#<CYCLIC-ITEM-STREAM #xF49521>
? (read-items z)
(A 1)
? (read-items z)
(2 3 B)
? (read-items z)
(4)
? (read-items z)
(5 C A 1)
? (read-items z)
(2 3)
? (read-items z)
(B 4 5)
... etc. ...
```

Schließlich kann jeder Item Stream über die Option `named` *<name>* einen Namen erhalten und so konkurrierend von mehreren Stellen aus gelesen werden. Die Funktion `motive`—abgekürzt `#@`*<name>*—referenziert dabei den benannten Stream.

```
? (items jill>
         (items ko fu ri sa no be ya ti
                in random
                for (items 3 1 2 4)
                named speak-japanese)
         jack>
         (items nope. huh? #@speak-japanese
                in random for 1)
         named play)
#<CYCLIC-ITEM-STREAM Play>
? (read-items #@play)
(JILL> NO SA RI JACK> HUH?)
? (read-items #@play)
(JILL> KO JACK> TI KO)
? (read-items #@play)
(JILL> FU BE YA FU JACK> NOPE.)
? (read-items #@play)
(JILL> FU BE FU JACK> FU)
```

Neben den genannten Merkmalen: Periodenlänge, Auswahlverfahren, Zählgrundlage und Name haben alle Item Streams einen definierten Datentyp (der nicht zu verwechseln ist mit Lisp–Datentypen) und akzeptieren über die Option `in` die Spezifikation eines vom Standard–Pattern `cycle` abweichenden Pattern Typs, so daß zusammenfassend folgende Optionen generell für alle Item Streams gelten:

`in` {*pattern*}
`for` {*integer* | *stream*}
`counting` {`periods` | `values`}
`traversing` {`depth` | `breadth`}
`named` {*name*}

3.3.2 Datentypen

Der spezifische Datentyp eines Item Streams wird über sein *Constructor Macro* festgelegt. `items` erstellt so einen Item Stream mit generischem (beliebigem) Datentyp. Diese Constructor Macros haben sämtlich eine Plural–Endung und sind nicht zu verwechseln mit etwaigen gleichnamigen Funktionen, die auf denselben Namen im Singular lauten. Beispielsweise erzeugt `notes` einen Item Stream vom Typ Note Stream, wohingegen note einer Zahl abhängig von der gerade aktuellen Skala (s. 3.5) einen Notennamen zuordnet!

Neben dem generischen Datentyp *Item* sind neun weitere Item–Stream–Datentypen in Common Music definiert. Mit diesen weiteren Typen sind dabei eine Reihe neuer, je nach Typ anwendbarer Optionen verbunden.

3.3.2.1 Note, Pitch und Degree Streams

`notes`, `pitches` und `degrees` erstellen Note, Pitch und Degree Streams, respektive. Dabei ist es ohne Bedeutung, welchen Datentyp die Elemente haben: Symbole werden konsistent als Tonnamen, ganze Zahlen als Tonhöhenstufen ('*Keynumbers*') und Kommazahlen oder Brüche als Frequenzen in Hz interpretiert. Werden Tonnamen ohne Oktavlagenzahl angegeben, erhalten sie die Oktavlage der zeitlich unmittelbar vorangegangenen symbolischen Tonhöhenangabe—die nicht zwingend im näheren lexikalischen Umfeld erfolgt sein muß!—oder, falls eine solche nicht existiert, den Initialwert der Variablen `cm::*standard-octave*`, 4. Daher wird `fs` anfanngs als `fs4`, im weiteren Verlauf aber als `fs5` (739.99 Hz, Degree 78), `b` higegen konstant als `b6` (1975.53 Hz, Degree 95) interpretiert:

```
? (read-items (notes 443.0579 30 fs f6 b g5))
(A4 FS1 FS4 F6 B6 G5)
? (read-items (pitches 443.0579 30 fs f6 b g5))
(443.0579 46.24929765929132 739.9887625486612 1396.9127692855886
1975.5329837758381 783.9907841609103)
? (read-items (degrees 443.0579 30 fs f6 b g5))
(69 30 78 89 95 79)
```

Alle drei Streams akzeptieren eine zusätzliche Option

`of` <*scale*>

mittels welcher die den internen Berechnungen des Item Streams zugrundeliegende Skala gewechselt werden kann. Wird keine Skala explizit angegeben, beziehen sich Tonhöhen– und Degree–Angaben auf den Wert der Variable `*standard-scale*`, die normalerweise die Skala *Standard-Chromatic-Scale* referenziert. Ist also beispielsweise die Skala *Pelog* definiert, ergibt sich für die Wechselbeziehungen zwischen Tonnamen, Stufen und Frequenzen ein reichlich ungewohntes Bild:

```
? (read-items (notes 11 pelog1 567.423 7 of (find-scale 'pelog)))
(DADA2 PELOG1 BEM1 NEM1)
? (read-items (degrees barang0 gulu0 nem2 bem1 lima1 of (find-scale
'pelog)))
(1 3 14 9 6)
? (read-items (pitches barang0 gulu0 nem2 bem1 lima1 of (find-scale
'pelog)))
(236.4721347825532 304.54811827742174 904.2177199471121 524.1586417873829
384.15011629720715)
```

Die Konversionsfunktionalität von Note, Pitch und Degree Streams steht übrigens auch außerhalb des Item–Stream–Protokolls zur Verfügung über die drei Funktionen

`note` <*scale degree reference*> &optional <*scale*>
`pitch` <*scale degree reference*> &optional <*scale*>
`degree` <*scale degree reference*> &optional <*scale*>

wie die folgenden Beispiele zeigen:

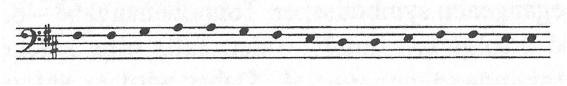
```
? (note (degree (item (pitches c4))))
C4
? (note (pitch 'barang1 (find-scale 'pelog)))
B4
```

3.3.2.2 Interval, Step und Series Streams

Im Gegensatz zu den Note, Pitch und Degree Streams, die *absolute* Werte erzeugen, erhalten Interval, Step und Series Streams ihre Resultate durch *relative* Berechnung. Ihre Elemente müssen daher numerisch gegeben sein. Am leichtesten darstellbar sind die Unterschiede ihrer Funktionsweise im Tonhöhenbereich: die beiden Streams

```
(intervals 4 4 5 7 7 5 4 2 0 0 2 4 4 2 2 from 'd3)
(steps 0 1 2 0 -2 -1 -2 -2 0 2 2 0 -2 0 from 'fs3 for 15)
```

generieren dieselbe (als Tonfolge interpretierte) Folge



Während der Interval Stream also jedes Datum als Intervallabstand von einem *fixen* Bezugspunkt auffaßt, konstruiert steps seine Werte über eine Differenzenfolge, lediglich von einem *Ursprung* ausgehend. Diese Arbeitsweise von Step Streams bringt es allerdings mit sich, daß sie notorisch ein Element weniger als zu erzeugende Items haben, weshalb ihre

Periodenlänge oft explizit angegeben werden muß. Alle drei Item Streams verwalten darüberhinaus natürlich gleichermaßen ganzzahlige wie nicht–ganzzahlige Daten.

Folgende Zusatzoptionen können für Interval, Step und Series Streams verwandt werden:

`of` <*scale*>
`from` {<*scale degree reference*> | <*stream*>}
`on` {<*scale degree reference*> | <*stream*>}
`initially-from` <*scale degree reference*>
`linked-to` <*name*>
`returning` {`note` | `pitch`}

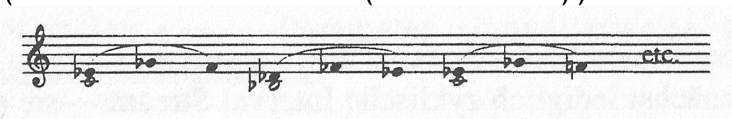
Die Option `of` erlaubt, wie oben für Note, Pitch und Degree Streams gezeigt, eine andere Skala als die von `*standard–scale*` referenzierte zugrundezulegen und mittels `returning` kann für die zurückgelieferten Werte ein bestimmter Datentyp (symbolische Tonnamen oder Frequenzen) gefordert werden.

`from`, `initially–from`, `linked–to` und `on` dagegen sind konkurrierende Optionen, die bestimmen, zu welchen Zeiten ein neuer, *externer* Bezugspunkt gewählt wird. Generell bezieht sich jede Periode erneut auf einen solchen Wert, der über eine der drei Optionen `from`, `linked–to` oder `on` gegeben sein kann. Mit der Option `initially–from` aber nimmt der Item Stream *nur anfänglich* einen solchen externen Wert als Ausgangspunkt, um sich dann im weiteren Verlauf bei Beginn jeder Periode nur noch auf seinen jeweils zuletzt erzeugten, *internen* Wert zu beziehen. Die Option `from` unterscheidet sich von `linked–to` darin, daß sie ihr Argument mit Beginn jeder Periode *evaluiert*, im Falle eines Item Streams also jeweils ein neues Element liest (und damit 'verbraucht'), während `linked–to` lediglich den gerade aktuellen Wert des referenzierten Item Streams zurückliefert. `on` schließlich evaluiert einen gegebenen Stream nicht nur zu Beginn einer Periode, sondern sozusagen *parallel*, mit jedem erzeugten Element.

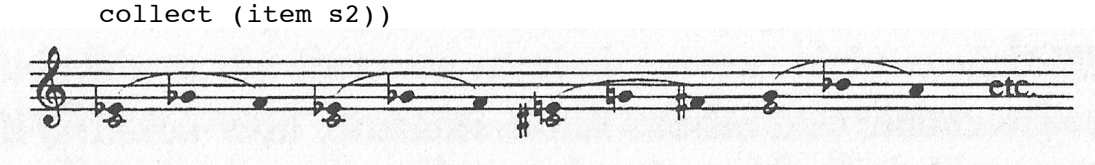
die folgende Reihe von Beispielen verdeutlicht diese Zusammenhänge von Interval bzw. Step Stream, Periodenlänge und je nach Option verschiedenen Bezugspunkten (Bezugspunkte sind hier durch 'weiße' Notenköpfe, von anderen Prozessen verbrauchte Werte eingeklammert sowie Perioden durch Bindebögen dargestellt):

```
(intervals 3 6 5 initially-from 60)
```

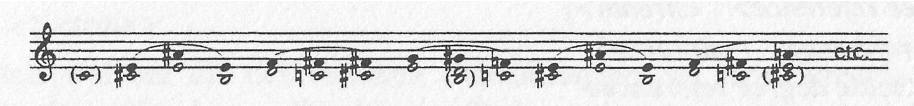
```
(intervals 3 6 5 from (items 60 58))
```



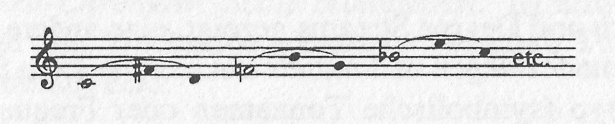

```
(loop for x below 12
      with s1 = (items 60 61 64 named offs)
      with s2 = (intervals 3 6 5 linked-to offs)
      do (when (= (mod x 4) 0) (item s1))
      collect (item s2))
```



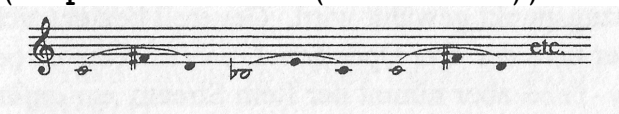
```
(loop for x below 15
      with s1 = (items 60 61 64 59 62 named offs)
      with s2 = (intervals 3 6 5 on #@offs)
      do (when (= (mod x 7) 0) (item s1))
      collect (item s2))
```



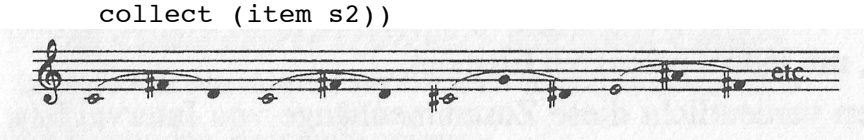

```
(steps 6 -4 3 initially-from 60)
```



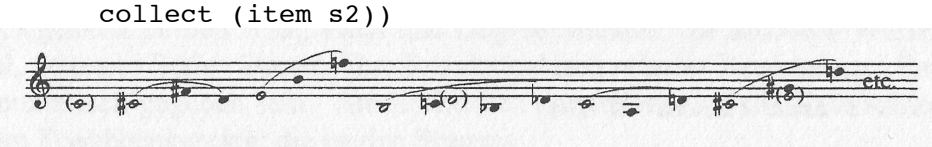

```
(steps 6 -4 3 from (items 60 58))
```




```
(loop for x below 12
      with s1 = (items 60 61 64 named offs)
      with s2 = (steps 6 -4 3 linked-to offs)
      do (when (= (mod x 4) 0) (item s1))
      collect (item s2))
```




```
(loop for x below 16
      with s1 = (items 60 61 64 59 62 named offs)
      with s2 = (steps (items 5 -4 2 -3 for 3)
                       (items 7 6 0)
                       (items 1 -2 3 0) on #@offs)
      do (when (= (mod x 7) 0) (item s1))
      collect (item s2))
```

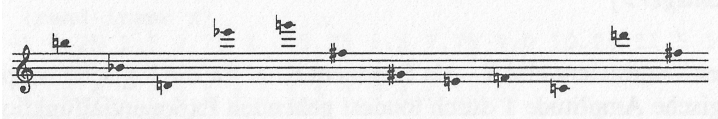


Ebenso wie Interval und Step Streams berechnen Series Streams ihre Elemente *relativ* zu einem Bezugswert. Series Streams sind dabei zunächst lediglich zyklische Interval Streams—sie sind in der Tat nur für den Standard–Pattern–Type `cycle` definiert—, implementieren jedoch dafür über die Funktionalität von Interval Streams hinaus einige Basisoperationen der Zwölftontechnik. Die auf Series Streams zusätzlich anwendbaren Optionen sind

`forming` {`:p` | `:i` | `:r` | `:ri` | <*stream*>}
`multiple` {<*integer*> | <*stream*>}
`modulus` {<*integer*> | <*stream*>}

Alle drei Optionen sind parallel verwendbar und nahezu selbsterklärend: über `forming` können Reihenformen (*prime*, *inverse*, *retrograde* bzw. *retrograde–inverse*), über `multiple` Skalierungen und über `modulus` Beschränkungen des Ambitus eingeführt werden.

```
(series 12 -1 3 from (notes b4 fs6 f4 c6)
        forming (items :p :ri :r :i)
        multiple (items 1 1 1 2)
        modulus (items nil nil 12 nil))
```



3.3.2.3 Rhythm Streams

Ähnlich wie Note, Pitch und Degree Streams Noten, Stufen bzw. Frequenzen, erzeugen Rhythm Streams aus ihren Daten—die dabei in verschiedenen Formaten vorliegen können—*Fließkommazahlenwerte* rythmischer Dauern.

```
? (read-items (rhythms q. s s te te te q+e e s 32nd 32nd e h w l))
(1.75 0.25 0.3333333333333333 0.3333333333333333 0.3333333333333333 1.5
0.5 0.25 0.125 0.125 0.5 2.0 4.0 8.0 16.0)
```

Darüberhinaus kennen Rhythm Streams die Zusatzoption

`tempo` {*<number>* | *<tempo function>*}

mittels welcher die Daten im Kontext einer bestimmten Tempoentwicklung interpretiert werden können. Je nach Grad ihrer Komplexität kann dabei auf unterschiedliche Notationen zurückgegriffen werden:

```
? (read-items (rhythms q. s s e e h+e e tempo 120))
(0.75 0.125 0.125 0.25 0.25 1.25 0.25)
? (read-items (rhythms q. s s e e h+e e
                       tempo (tempo from 60 to 120 in 6)))
(1.3125 0.21354166666666669 0.20833333333333334 0.39583333333333337 0.375
1.354166666666667 0.25)
? (read-items (rhythms q. s s e e h+e e tempo (tempo 0 60 3 76 6 120)))
(1.3421052631578947 0.2192982456140351 0.2149122807017544
0.41228070175438597 0.39473684210526316 1.3706140350877194 0.25)
```

Ineinander geschachtelte Rhythm Streams bleiben jedoch stets unbeinflußt von Tempoangaben der anderen Streams und können so separate Tempokurven erhalten.

Ebenfalls wie bei Note, Pitch und Degree Streams kann die Konversion von rythmischer Notierung nach absoluten Dauernwerten auch unabhängig von Rhythm Streams mit der Funktion

`rhythm` *<rhythm>* &optional *<tempo>*

erfolgen. Da durch `rhythm` jedoch lediglich einzelne Rythmen umgerechnet werden, sind für den optionalen `tempo`–Parameter sinnvollerweise keine Tempofunktionen, sondern nur Metronomzahlen zugelassen.

3.3.2.4 Amplitude Streams

Für den Bereich der Lautstärken bieten Amplitude Streams eine ähnliche Funktionalität. Auch hier werden in erster Linie numerische oder symbolische Lautstärkenbezeichnungen in absolute Werte übersetzt. Diesen Amplitudenwerten liegt jedoch, um sie unabhängig vom Ausgabemedium zu halten, ein Konzept *logischer* Amplitude zugrunde: sie sind normalisierte (d.h., zwischen 0 und 1 liegende Werte, die über eine spezielle—wenn nicht explizit angegeben, lineare—Funktion sowie optionale Begrenzungswerte auf den ebenfalls normalisierten Bereich *roher* Amplituden abgebildet werden. Erst die jeweilige Output Syntax schließlich übersetzt diese rohen Werte in endgültige Lautstärkeangaben.

Entsprechend dieser Übersetzung von logischen in rohe Amplituden akzeptieren Amplitude Streams die Zusatzoptionen

`softest` {*<float>* | *<integer>*}
`loudest` {*<float>* | *<integer>*}
`power` *<float>*

wobei der Wert von power—normalerweise 1—als Exponent einer für die logische Amplitude 0 durch softest und für die logische Amplitude 1 durch loudest gehenden Exponentialfunktion dient. Größere Werte für power treiben daher die Amplituden aus dem Mittelbereich nach außen, während umgekehrt kleinere die Amplituden im Mittelbereich verdichten:

```
? (read-items (amplitudes pppp p f ffff))
(0.1 0.4 0.7 1.0)
? (read-items (amplitudes pppp p f ffff softest .25 loudest .75))
(0.3 0.45 0.6 0.75)
? (read-items (amplitudes pppp p f ffff power 3))
(0.0010000000000000002 0.06400000000000002 0.3429999999999999 1.0)
? (read-items (amplitudes pppp p f ffff power 1/3))
(0.464158883361278 0.7368062997280773 0.8879040017426006 1.0)
? (read-items (amplitudes pppp p f ffff softest .25 loudest .75 power 2))
(0.255 0.33 0.495 0.75)
```

Die Syntax der Konversionsfunktionalität von Amplitude Streams entsprechende Funktion lautet

`amplitude` *<amplitude>* &key *<softest>* *<loudest>* *<power>*

Die beiden Hilfsmakros

`crescendo` &body *<body>*
`diminuendo` &body *<body>*

schließlich erstellen auf intuitive Weise zyklische Amplitude Streams mit auf– oder absteigenden Amplitudenwerten:

```
? (read-items (crescendo from pppp to ffff in 6))
(0.1 0.28 0.46 0.64 0.82 1.0)
? (read-items (diminuendo from ff to p in 7))
(0.8 0.7333333333333333 0.6666666666666666 0.6 0.5333333333333333
0.4666666666666667 0.4)
```

3.3.2.5 Number Streams

Number Streams schließlich erlauben über eine der Schrittfolgenspezifikation in Common Lisp's `loop`–Makro analoge Syntax die unaufwendige Generierung von diskreten Zahlenfolgen und –mengen. Wegen ihrer inhärent generativen Natur sind sie dabei allerdings nur für die Pattern Types `cycle` und `random` implementiert. Von den für Number Streams spezifischen Zusatzoptionen

`from` {*<number>* | *<stream>*}
`to` {*<number>* | *<stream>*}
`below` {*<number>* | *<stream>*}
`downto` {*<number>* | *<stream>*}
`above` {*<number>* | *<stream>*}
`by` {*<number>* | *<stream>*}

muß lediglich exakt eine Zielangabe (`to`, `downto`, `below` oder `above`) vorhanden sein—ein von 0 abweichender Startpunkt oder eine andere als die Standard–Schrittweite 1 können davon unabhängig jederzeit mittels der `for`– bzw. `by`–Option bestimmt werden.

```
(setf x (numbers from (numbers above -5)
                 to (numbers from 20 downto 0 by 5)
                 by (numbers from 1 to 2 by .25)))
? (read-items x)
(0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20)
? (read-items x)
(-1 0.25 1.5 2.75 4.0 5.25 6.5 7.75 9.0 10.25 11.5 12.75 14.0)
? (read-items x)
(-2 -0.5 1.0 2.5 4.0 5.5 7.0 8.5 10.0)
? (read-items x)
(-3 -1.25 0.5 2.25 4.0)
? (read-items x)
(-4 -2.0 0.0)
```

3.3.3 Pattern Types

Mit wenigen Einschränkungen (Series und Number Streams betreffend) kann unabhängig vom Item–Stream–Typ ein beliebiger Pattern Type über die generelle Option `in` angegeben werden. Diese Pattern Types akzeptieren ebenfalls oft—ähnlich wie die verschiedenen Data Types—speziell auf ihre Funktionalituat hin zugeschnittene Zusatzoptionen.

3.3.3.1 Cyclic, Sequential und Accumulating Streams

Zu den einfachsten Pattern Types gehöhren der Standard–Pattern–Type `cycle` sowie die Typen `sequence` und `accumulation`. Anstatt ihren Datenvorrat wie zyklische Streams nach Durchlauf aller Elemente zu reinitialisieren, *repetieren* Sequential Streams schlicht endlos ihr zuletzt ausgegebenes Element:

```
? (setf x (items 1 2 3 4 5 6 7 in sequence for 4))
#<SEQUENTIAL-ITEM-STREAM #xB80511>
? (read-items x)
(1 2 3 4)
? (read-items x)
(5 6 7 7)
? (read-items x)
(7 7 7 7)
? (read-items x)
(7 7 7 7)
... etc. ...
```

Accumulating Streams dagegen sind in dem Sinn zyklisch, daß sie ihr Gesamtmuster—wenn nicht über die `for`–option anders angegeben, ihre Periode—immer gleichbleibend wiederholen. Dieses Gesamtmuster selbst jedoch wird aus den gegebenen Daten in kumulierender Weise aufgebaut.

```
? (read-items (items 1 2 3 4 5 6 7 in accumulation))
(1 1 2 1 2 3 1 2 3 4 1 2 3 4 5 1 2 3 4 5 6 1 2 3 4 5 6 7)
```

Cyclic, Sequential und Accumulating Streams haben keine speziellen Zusatzoptionen.

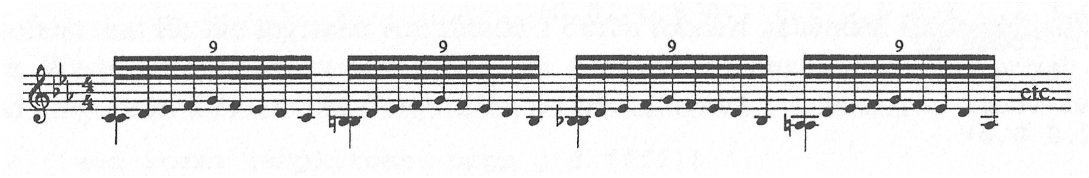
3.3.3.2 Palindromic Streams

Ebenfalls noch zu den einfachen Pattern Types zählen Palindromic Streams. Das von ihnen erzeugte Muster wird erzeugt durch wiederholte Ausgabe der Daten mit Indizes 1, …, *n*, [*n*,] …, 1, [1,] …, usf. Die Option

`elided` {`yes` | `no` | `start` | `end` | `left` | `right`}

kontrolliert dabei, *ob* alle (`yes` oder `no`) bzw. *welche* (`start` oder `end` bzw. `left` oder `right`) Randelemente der Datenliste von einer Wiederholung an den Nahtstellen des Patterns ausgeschlossen werden.

```
(notes (notes c4 c b3 b bf bf a a for 1) d4 ef f g
       in palindrome elided end)
```



3.3.3.3 Rotational Streams

Rotational Streams führen iterativ *Versetzungsoperationen* über je zwei Elemente der Datenliste durch, in Abhängigkeit der über

`change` {*<list>* | *<stream>*}

gegebenen Direktiven. Die maximal vier Werte einer an die `change`–Option übergebenen Liste werden von Rotational Streams entsprechend ihrer Position als Index des Elementes, bei dem die Vertauschung beginnt, Schrittbreite der Iteration, Differenz der Indizes der beiden zu vertauschenden Elemente sowie Index des Elements, mit dem die Iteration abbricht interpretiert. Anstelle einer solchen Liste kann auch das Makro

`changes` &rest *args*

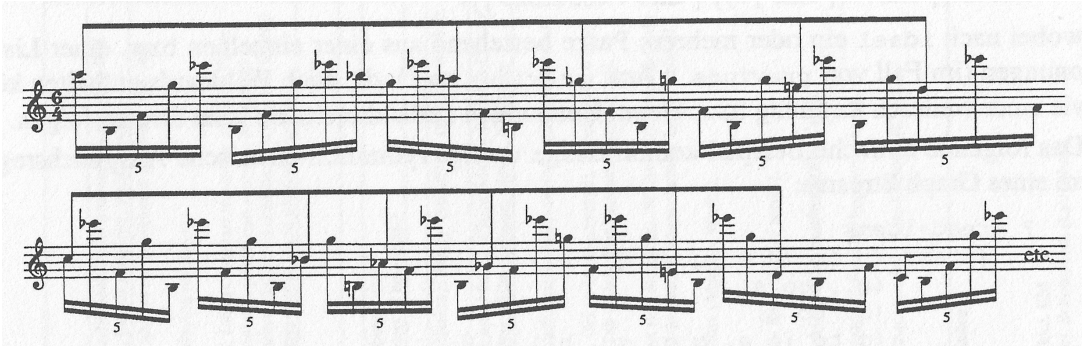
mit einer beliebigen Kombination der Parameternamen `start`, `step`, `width` oder `end` eingesetzt werden. Wird darüberhinaus einem der Parameter von `changes` eine Liste übergeben, so nimmt er nacheinander reihum die Werte dieser Liste an (s. dazu das Beispiel unten).

Sind über die Option `change` keine anderen Werte angegeben, so wird für `start` der Standardwert 0, für `step` wie für `width` 1 und für `end` `t`, d.h., das letzte Element angenommen. Mit diesen Voreinstellungen erzeugt der Stream eine Folge von Mustern, während derer der Datenvorrat jeweils einen Schritt weiter rotiert wird:

```
? (setf x (items * 1 2 3 4 in rotation))
#<ROTATIONAL-ITEM-STREAM #xBA1101>
? (loop repeat 6 collect (read-items x))
((* 1 2 3 4) (1 2 3 4 *) (2 3 4 * 1) (3 4 * 1 2) (4 * 1 2 3) (* 1 2 3 4))
```

Die von Schritt zu Schritt durchgeführte iterative Vertauschungsoperation kann dabei, vor allem, wenn der Stream etwa von eingeschlossenen Streams Gebrauch macht, sehr schnell zu komplexen Folgen mit langen wiederholungsfreien Perioden führen.

```
(notes (steps -2 initially-from 'c6 returning note)
       b3 f4 g5 ef6
       in rotation
       change (changes width '(1 2)))
```



3.3.3.4 Random und Heap Streams

Random und Heap Streams führen—wie bereits in vielen Beispielen zu sehen war—eine pseudo–zufällige Auswahl (`random`) respektive Reihung (`heap`) der Daten durch, ausgehend von einem über die Option

`using` *<random state object>*

bestimmbaren Common Lisp *Random–State*–Objekt.

Darüberhinaus erweitern Random Streams die Syntax von Item–Spezifikationen um

{*<item>* | *<stream>*} &key (`weight` 1) (`min` 1) (`max` `nil`)

Auf diese Weise können sowohl relative Gewichtung als auch minimale und maximale Länge von ausschließlich aus Elementen dieses Items oder Item Streams bestehenden Sequenzen kontrolliert werden.

3.3.3.5 Graph Streams

Mittels Graph Streams können komplexe Folgerelationen zwischen den Daten des Streams bis hin zu Markovprozessen ausgedrückt werden. Die Daten werden dabei als *Knotenpunkte* mit optionaler *Kennung* sowie einer Angabe möglicher Nachfolger aufgefaßt. Ähnlich wie Random Streams erweitern so auch Graph Streams die Syntax von Item–Spezifikationen um

{*<item>* | *<stream>*} &key (`id` <item>) (`to` `nil`)

Die optionale `id` dient dem Knoten als Kennung und kann ein beliebiges Symbol, Keyword oder eine Zahl sein, während `to` in Form einer einzelnen oder einer Liste von Kennungen oder als Item Stream von Kennungen gegeben werden *muß*. Die Zusatzoptionen für Graph Streams sind:

`with` {*<function>*}
`previous` {*<number>* | *<id list>*}

Stehen durch Angabe einer Liste für `to` mehrere Nachfolger zur Auswahl, wird unter ihnen entweder zufällig oder mittels einer über die Option `with` bestimmten Auswahlfunktion gewählt. Die Zahl der an diese Funktion übergebenen *zuletzt gewählten* Elemente läßt sich dabei über `previous` entweder als Nummer oder als Liste von Kennungen—die damit zugleich auch die Initialwerte dieser vorangegangenen Elemente bilden—angeben.

Wird—wie z.B. in Markovprozessen—für jeden Knoten eine separate auf Wahrscheinlichkeiten beruhende Ausfwahlfunktion benötigt, bietet das Makro `idsel` eine bequeme Möglichkeit, die dieser Auswahl zugrundeliegenden Entscheidungsmerkmale in einfacher Form und mittels Item Streams zu kodifizieren. `idsel` hat die Syntax

`idsel` [{*<id>* | ({*<id>*}+)} {*<id>* | *<stream>*}]+

wobei nach `idsel` ein oder mehrere Paare bestehend aus einer einzelnen bzw. einer Liste von Kennungen (im Fall von `previuos 1` bzw. `previous` *n>1*), die auch Wildcards enthalten können, sowie einer weiteren Kennung bzw. einem Kennungen zurückliefernden Item Stream folgen.

Das folgende einfache Beispiel kodiert einige wenige rythmisch–metrische Ausgleichsregeln in Form eines Graph Streams:

```
? (read-items
   (items (h to (e e. q q. q.. h))
          (q.. to s)
          (q. to e)
          (q to (e e. q q. q.. h))
          (e. to s)
          (e to (idsel e (items e e. q q. q.. h in random)
                       * e))
          (s to (e e. q q. q.. h))
          in graph for 20 previous (q)))
(H Q.. S Q. E E Q E E E Q E. S Q. E E E. S Q. E)
```

3.3.3.6 Functional Streams

Mit Functional Item Streams schließlich bietet das System eine Möglichkeit, selbstdefinierte Funktionen in das Item–Stream–Protokoll einzubinden. Sie erhalten keine expliziten Daten sondern lediglich eine—benannte oder anonyme, sog. ‘Lambda–’—Funktion, die über die Option

`with` *<function>*

eingeführt wird. Soll die Funktion Argumente erhalten, sind sie mittels der Option

`args` *<arglist>*

explizit zu deklarieren. Das Resultat der Funktion bildet die neue Periode des Functional Streams; sie kann daher sowohl ein einzelnes als auch eine Liste von Elementen, darf jedoch keinesfalls die leere Liste (`nil`) zurückliefern.

```
? (let ((x (items in function with
                  #'(lambda ()
                      (list (random 10) (random 10))))))
    (loop repeat 4
          collect (read-items x)))
((2 5) (5 3) (5 4) (9 2))
```

Fortsetzung in den Mitteilungen_15:

3.4 Bearbeitung im Editor
3.5 Skalen und Scale References
3.6 Input und Output
3.7 Die Programmierung von Stella
3.8 Erweiterung des Systems
4. Quellcode und Installation

Internationaler Kalender Elektroakustischer Musik 1994 (ab September 1994)

September 1994

2.	Jour fix 20 Uhr Akademie der Künste	Luisenstr. 58; 10117 Berlin	offene Gesprächsrunde im Akademiestudio unter Leitung von Georg Katzer
2. - 4.	Kunstverein Giannozzo, Rolf Langebartels; Informationen: 030 - 321 77 83	im Kloster Plasy, Tschechien	Thema „Feedback, das Phänomen der Rückkopplung in Kunst und Wissenschaft". Reise von Berlin nach Plasy und zurück mit dem Bus. Performances / Aktionen von Marie Kawazu, Rolf Langebartels, Udo Idelberger, Tom Johnson, Benoit Maubrey, Akio Suzuki
2. - 11.	Multiple Arts Festival Stichting Intro 7 locations in the town	St. Maartenspoort 2 NL - 6221 BA Maastricht fax: 0031 43 - 25 74 88	Thema „Music and visual arts". Schwerpunkte „young artist´s concert-and-image circuit" and „special commissioned projekts" (in- and outdoors installations). **Performances**: 2. - 4.9. Johann Schmidt-Sistermanns (Klangort); Martin Riches (flute playing machine) 10., 11.9. Hans van Kookwijk (Bambuso Sonoro) **Installations**: Christof Schläger (Geräusch aus den 4 Weltgegenden), Stichting Logos (RAdar), Johan Goedhart (concrete sounds), Pierre Bastien (Mecanium), Bosch & Simons (de trajectversterker), Imogen Stidworthy , Habs van Koolwijk (Bambuso sonoro), Jacqueline Moors, Anders / Berger / Banze / Mittelstädt (lumen), Sebastian Jochum (Luft holen), Anna Anders (Schützen schieben), Krzycztof Knittel (the legs), Paul Devens, Guy Bleus (talking chairs) sowie Robin Minard (silent music) und Martin Riches (flute playing machine)
3. - 10.	2. internat. Akademie für Neue Komposition AVANTGARDE Musikschule	A - 6130 Schwaz	Ausstellung : Musikgrafiken von Boguslaw Schaeffer und Roman Haubenstock-Ramati. Konzert Marek Choloniewski (open air performance)
5. - 10.	Nordische Musiktage	DK Kopenhagen	auch Programm mit EM. Infos: +45-33-143219
5. - 11.	Gaudeamus Music Week Gaudeamus Foundation	Swammerdamstraat 38 NL 1091 RV Amsterdam	EM gespielt von: Kyriakides (5.9.), Michael Obst (Miroirs, 9.9.), Tim Howle und Jos Zwaanenburg (10.9.) General meeting: IAMIC (int. ass. of music inform. centres), ECPNM (europ. confer. of promoters of new mus.)
9. - 11.	Randspiele St. Annen Kirche	Zepernick bei Berlin	
10. - 18.	Colourscape festival	GB - Clapham Common	Melvyn Poore (Music-Theater); Tim Souster (Hambledon); B. Oliver / A. Visser (installation); Barry Andersen (several works); Stockhausen (solo), J. Harrison (Monodies); Emmerson (Time Past); Moore (Sieve); Guy (EOS)
10. - 14.	1st Brazilian Meeting for EM LME , Uni. Brasilia	Brasilia DF	Foundation of Brazilian Society of EM; 35 years of Brazilian EM. 5 concerts with EM by Lintz-Maues, Chico Mello, Rodolfo Caesar, Paulo Chagas and many other composers from Brazil. Open air concert: EM von Mannis, Escobar, Leite, Antunes, Neves, F. Richter
12. - 17.	ICMC ICMA / DIEM E-mail: ICMC94@daimi.aau.dk	Musikhuset Aarhus DK 8000 Aarhus C fax: +45-8612 0807 **Concert 1. 13.9.** **Concert 2.,14.9.** **Concert 3.,** **Concert 4., 15.9.** **Ballet 5.** **Concert 6.** **Concert 7., 16.9.** **Concert 8.** **Concert 9., 17.9.** **Concert 10.** **Concert 11.**	International Computer Music Conference 1994. Focus: The Human Touch. **Vorprogramm**: 12.9.: DSP Perry Cook); KYMA (Carla Scaletti); 13.9.: Aesthetics (Wishart), Interactive Automata (George Lewis). **Installations** by P. Bosch & S. Simons, Grølund & Tin, Kean, Svalkeapää & Nousiainen. **Cikada** Oslo: Montague (Silence…), Truax (Sequence of…), Smith (Flux), Cipriani (Visibili), DeLio (Anti-paysage), Thoresen (Ab Uno) Pecquet (Cello&Co), Taube (Gloriette for Cage), Eldènius (Norwegian fragments), Mowitz (I Tak Dalej), Dunkelman (Rituellipses), Tedde (Cello voce), Nuñez (Jurel), D. Smalley (Piano Nets) **Multimedia**: Wallin (Yo), Simura (Tikukan no utyu II), Cadoz / Luciani (Esquisses), Matthews (in emptiness), Waisvisz (Faustos Schrei) Hamel (Obsessed again…), Ceccarelli (Birds), Lippe (Music for cl. & ISPW), Lundèn (Noises), Morales (Cempaxuchitl), Lee (YunMu 3), Tanaka (Overbow) Royal Danish **Ballet**: Ainger (Lament), Wishart (Tongues of Fire), Chen (Huang Zhong), Pinkston (Margo's World), Parmerud (Jeux imaginaire), Karpen (Terra infirma) Fireworks: Dhomont (Artifices), Parmerud & Blomqvist (Trio target) Phelps (Sax houses), Gobeil (Le vertige inconnu), K. Norman (Trilling wire), Stammen (Tuva!), Rahn (Sea of souls), Gawad (Taqaseem 2), Penrose (Manwich), Alford (Image) **Athelas** Ensemble & **Karttunen** (Cello): Jaffe (7 wonders), Saariaho (Amers), White (Winter Aconites), Frounberg (SHâTaLè), Fredrics (The Raven´s Kiss), Højsgaard (Nocturne) Pennycook (Praescio 3), Roy (Crystal Music I), Nelson (Waves of Refraction), Normandeau (Spleen), Favilla (Improvisations), Austin (Rompido), Viñao (El Simurgh, book 2) Aarhus Symphony **Orchestra**, **Mead** (Piano): Giroudon/Dorival (Double 3), Bevelander (Synthecisms #4), Kats-Chernin (Clocks), Lanza (Piano concerto), Hauksson (Bells of earth) Computer **Jazz**: Strange (Velocity), Berg (Loops), Nilsson (Random), Fuzzy (ICMC Surprise)
15. - 24.	Warschauer Herbst	Rynek Starego Miasta 27 PL - 00-272 Warszawa fon-fax: +48-22-310607	15.9. „Electric Night I": Lansky (Smalltalk), Cienciala (Epiphora II), Calon (Portrait d'un visiteur), Bien (Turdidae), Fomina (Expulsion,…), Parmerud (Stringquartet) 16.9. Jean Claude Eloy: Anâhata 1, Erkos 17.9. Jean Claude Eloy: Butsumyoe, Sappho Hiketis, Yo-In 18.9. Krupowicz (Fin de siècle), Kulenty (Violin concerto), Jan Pilch spielt diverse Werke, incl. EM 19.9. „Electric Night II": Wingate (Ode to the South-Facing Form), Czaja (Jeszcze zmierzch), Schottstaedt (Leviathan), Zwedberg (And I killed him twice), Takenaka (Memory of the Universe) 20.9. Audio-Art Concert mit Monahan, Moss, M. Choloniewski, Barber, Raes&Darge, Rickels&van Brakel 23.9. Cikada Ensemble spielt u.a. Tristan Murail (Allégories)

noch September

15. - 18.	7. Tage Neuer Musik: Klänge Sprache Zeichen Klangprojekte Weimar	Schillerstr. 11a; 99423 Weimar	15.9. Harry Sparnaay Trio (Lachenmann, Tutschku, Berio, de Leeuw) 16.9. Ensemble für intuitive Musik führt Ernst Jandl auf 17.9. Berio, Buser, Kagel, Huber, Tutschku, Stockhausen, Cage 18.9. Ensemble für intuitive Musik „in memory of the Konstruktivistenkongress“ 1922 (DADA)
16.	44. Berliner Festwochen Kammersaal	PhilharmonieBerlin	Heinz Holliger: Scardanelli-Zyklus mit Ensemble Modern
17.	KölnMusik	Philharmonie Köln	Heinz Holliger: Scardanelli-Zyklus mit Ensemble Modern
21.	10th festival international; CDMC Madrid	E - Alicante	Phil Corner; LIEM-CDMC - Konzert
21. - 24.	sound and space	Stockholm	Electronic Music Festival XVI
23. - 25.	Concorso Russolo	I - 21110 Varese	Festival und Wettbewerb
23.	Musica 94	2 rue d‘Ingwiller F - 67000 Strasbourg	23. - 25.9. Saariaho (…à la Fumée, Nymphea, Solar, Amers, Jardin secret II, Gates); Lindberg (Corrente II) 5. - 6.10. Heiner Goebbels (Befreiung des Prometheus) 4. - 5.10. Bob Ashley (4 operas: improvement, Don leaves Linda, now Eleanor´s Ifea, foreign experiences)
24.	Berwald Hall / SR	Stockholm	Erik Mikael Karlsson (Figurer i fyra kappor)
24., 25.	Festival of Musical Action	Vilnius, Litauen	Beziehung zwischen Musik und anderen Kunstformen (Klanginstallationen, Video, …)
25.9. - 5.10.	Workshop für max 10 Personen Kursgebühr erfragen	Ritterstr. 42 D 76 137 Karlsruhe Tel.: 0721 - 9340300	„Algorithmische Komposition“, Leitung Rick Taube, Mitarbeit Tobias Kunze. Themen u.a. Patternbeschreibung, Scheduling Strategien, Kontrolle von Zufallsprozessen, Score Editing. Unterricht basiert auf „Common Music“ und „Stella“ mit Unterstützung von Music Kit, Common Lisp Music, CSound, MIDI. Arbeit an NeXT und Mac.
28. -	8th contemporary music; LIMCA	F - 32000 Auch	L‘Itineraire plays Mâche, Levinas, Harvey, Agobert, Grisey, Murail, Hervé, Verin, Alvarez, Vaggione

Oktober 1994

1. **Deadline**	Musica Nova ´94 Wettbewerb Bekanntgabe der Juryentscheidung am 1.11.94	Spolecnost pro elektroakustickou hudbu MUSICA NOVA ´94 Radlická 99 ; 150 00 Praha 5 Czech Republic	Regeln für den internationalen Wettbewerb für die beste elektroakustische Musik. Dauer beliebig, entstanden nach dem 1.1.1990, Abgabe des Tonträgers in der technisch besten Form Kategorie **A**. Kompositionen mit autonomer künstlicher EM. Kategorie **B**. Multimedia Projekte, die EM nutzen (audiovisuell, Bühne) Jeder darf maximal ein Werk je Kategorie einreichen. Jurymitglieder sind von der Teilnahme ausgenommen. Eine internationale Jury ist noch unbekannt, wird aber aus international anerkannten Persönlichkeiten bestehen. Die Einreichung kann namentlich oder anonym erfolgen. Anlagen: technische Dokumentation mit Partitur, Ausführungsvorschriften, szenische Vorgaben …Die eingereichten Bänder verbleiben im Archiv der SEAM. Mit der Einreichung sind die Aufführungsrechte auch im Rundfunk anerkannt.. Je Kategorie ist ein Hauptpreis in Höhe von 10.000 Kc, 2. Preis von 7.000 Kc vorgesehen; auch Sendung.
1. - 8.	World Music Days 1994 Svenska Rikskonserter	Box 1225 S - 111 82 Stockholm FAX.: (+46) 8 - 6760018	Jury: Magnus Andersson, Franco Donatoni, Michael Finnissy, Helmut Lachenmann, Kaija Saariaho.(siehe 1.4.1993 **Deadline**). Installationen in Fylkingen von D. Pignon, Doukkali, Tankred. Konzerte mit EM: 1.10. opening concert: Dror Feiler, Sten Hanson, Anders Hillborg, Karin Rehnqvist 4.10. Jonathan Harvey (Forms of Emptiness), Javier Alvarez, Bo Rydberg (LINK/Sequences), Gabriela Ortiz 6.10. Gilles Gobeil (Le vestige inconnu), Philippe Leroux, Tadeucz Wielecki 7.10 Henri Pousseur (Scambi), Luciano Berio (Perspectives)
1. - 16.	Under Cover; BMUK; 0043 3682 - 23250	A - Stainach im Ennstal	Improvisationen mit Anthony Braxton, Joëlle Leandre, Tom Cora, Wolfgang Mitterer, Burkhard Stangl. Max Nagl, Samm Bennett …
ab3.	Computer Music and Composition Les Ateliers UPIC FAX & Tel.: (+33) 1 - 60139339	5, allée de Nantes F - 91300 Massy	Angebot eines 2-semestrigen Programms mit Hochschul-Profil. Dozenten: Curtis Roads, Julio Estrada, Gerard Pape, Brigitte Robindoré, Iannis Xenakis, Michel Philippot. Kurse: Computer Music; Electronic Music Studio Techniques; Compositional Theory and Practice. Vorhandenes Equipment u.a. : UPIC, Axcel resynthesizer, ProTools, AKAI DD1000. Bis max. 10 Teilnehmer werden zugelassen.
5.	ZKM in der Fabrik „Bilocation“	Lorenzstr. Karlsruhe	Man Ray´s Stummfilm „L‘étoile de mer“ mit Musik von Johannes Goebel; Heinrich Taube (Swansong), Video- und Computeranimationswerke von Yamaguchi, Callas, Lanz
6. - 9.	Workshop für max 10 Personen Kursgebühr erfragen: Tel.: 0721 - 9340300	Ritterstr. 42 D 76 137 Karlsruhe	„Einführung in digitale Klangerzeugung“, Leitung Pierre Dutilleux. Themen: FM, add. und sbr. Synthese, lineare und nichtlineare Verwandlungen (Shifting, Phase Vocoder, Physical Modelling). Arbeit an NeXT-Computern mit ISPW-Karte und MAX-Programm.
6. - 9.	Musica verticale; via Gentile da Mogliano	I - 00176 Rom	
7.	Jour fix 20 Uhr, Akademie der Künste	Luisenstr. 58; Berlin	offene Gesprächsrunde im Akademiestudio unter Leitung von Georg Katzer
8., 9. 15., 16.	5 Jahre ZeM – Fest	PH Freiburg Steinhalle Emmendingen	Diskussion & Demos, Vorträge von Ludwig Rehberg, Joachim Stange, Vorführung Synthesizer Ausstellung / Performances von Weinhold & Schneider, Löhle; Konzert (Birg, Czech, Fiedler, Schneider)
9. - 16.	8. Dresdner Tage für zeitgenössische Musik Dresdner Zentrum für zeitgenössische Musik	Schevenstr. 17 D 01326 Dresden	9.10. profectio-initiative Freiburg spielt EM von Hummel, Rubin, Oung Lee, M. Schmidt, Breitenfeld 11.10. Josef Anton Riedl: Musik / Film / Dia / Licht - Multimedia-Programm 13.10. „Achsen des Jazz“, es spielen: Jeffrey Burns und „avant-garde trio Murilag“ (Waibel, Krieger, Agnesen) 16.10. „Electronics“: Wilfried Jentzsch (PAYSAGES SM und PC), Denis Lorrain (black it stood as night), Georg Katzer (Dialog imaginaire II), Magdalena Dlugozc (at the roots), Ensemble Modern spielt Musik von H. Goebbels (Herakles 2, …)
10. - 15.	encontre de compositors; Area Acustica	Palma de Mallorca	14.10.: Adolfo Nuñez
10. 24.	open air concert; LME , Uni. Brasilia Inst. de artes	70919-970 Brasilia DF	EM von Alice Shields , Ricardo dal Farra, Trevor Wishart J. Sermilä, Rune Lindblad, Edgar Fröse, Luciano Berio (Visage)

noch Oktober

13. - 15.	Ultima Festival Toftesgatan 69	N - 0552 Oslo 5	Mats Persson & Kristine Scholz & Brunson : Stockhausen (Mantra) Karlsson & Rabben & Enström: Stockhausen (Kontakte)
14. - 16.	Donaueschinger Musiktage fax: 0771-857228	78166 Donaueschingen	Musikmaschinen von : Stefan von Huene, Ed Osborn, Jackes Remus, Martin Riches, Horst Rickels, Peter Vogel, Harry de Witt, Ad van Buren, Pierre Bastien, Frederic le Junter, Gordon Monahan, Trimpin 14.10. Konzert mit dem Player piano (Hocker): G. Ligeti, Carlos Sandoval, Hans Haass, C. Nancarrow 15.&16.10. „Akustische Spielformen“: Karl-Scuka-Preis des SWF 1994
17. - 30.	Tage alter & neuer Musik	Bonn	vier Klangaktionen und „Geburtstags-Konzert“ (65) von und für Josef Anton Riedl
19.	discoveries Xa Northern College	GB - Aberdeen	Settel (Skweeit-chupp), Liffen (Sweeping …), Storie (Melwat), Adkins (Clothed…), Graebner, Smalley (Pentes)
20. - 28.	Tour Ashley Holland; De Ijsbreker e.a.	1091 EC Amsterdam	Robert Ashley (el/Aficionado) also in Groningen, Maastricht, Den Bosch, Rotterdam, Hague, Middleburg
29.	Medien-Nacht des 13. Synthesizer Musik Festivals; Neue Akademie Braunschweig	Landesmuseum Braunschweig	Preisträgern des 5. Wettbewerbes für Synthesizer- und Computermusik: „ART-VIDEO“ von Joan Pueyo, Peter Callas, Bériou; „konzertantes MIDI-Planetarium“ von W. M. Stroh und R. Schimmelpfennig; „Mimen-Klänge“ von Dieter Salbert und Andreas Haase.
30. **Deadline**	22 internationaler Wettbewerb für EM GMEB	BP 39 F - 18001 Bourges Cedex Tel.: 0033 48204187	Regeln und Organisation wie im Jahre 1993 (jedoch nicht im Rahmen des Synthese-Festivals); Kategorien: „Residence“ (für junge Komponisten, Studioaufenthalt); „Quadrivium“ (für Komponisten älter als 25); „Magisterium“ (für diejenigen, die seit mindestens 20 Jahren aktiv in der EM sind); „Le Puy“ (EM, die nicht für das Konzert gedacht ist) mit den 4 Themen „Musik und Humor“, „circumstantial music“, „EM für Jugendliche“, „EM für Tanz“.
31. **Deadline**	3. Wettbewerb Musik Kreativ	Schumannstr. 9 D - 60 325 Frankfurt	„Elektronik“-Wettbewerb für Profis bis Amateure jeden Alters
	Irish museum of modern art	box 3933, Dublin 8;	Audio Art Radio Transmission Projekt seeks tapes or CD´s

November 1994

4.	Jour fix 20 Uhr; Akademie der Künste	Luisenstr. 58; Berlin	offene Gesprächsrunde im Akademiestudio unter Leitung von Georg Katzer
4. - 14.	International Composers´ Workshop Foundation Gaudeamus	Swammerdamstraat 38 NL 1091 RV Amsterdam	Thema: world musical traditions & contemporary composition. Leitung: Ton de Leeuw, Nigel Osborn, Dimiter Christoff. Für aktive Teilnehmer (350 Gulden) und Zuhörer (175 Gulden). **Deadline**: 31.7.94
7. - 9.	ICAD´94 Santa Fe Institute FAX 001 - (914) 424-3467 E-mail: kramer@santafe.edu	1660 Old Pecos Trail Santa Fe, New Mexico 87051 USA	International Conference on Auditory Display. Topic areas : Auditory exploration of data via sonification (data-controlled sound) and audification (audible playback of data samples); Real-time monitoring of multivariate data; Sound in immersive interfaces (Virtual Reality) and Teleoperation; Perceptual issues in Auditory Display; Sound in generalized computer interfaces; Technologies supporting Auditory Display creation; Data handling and sound synthesis for Auditory Display systems; Applications of Auditory Display.
7. und 21.	open air concert LME , Uni. Brasilia; Inst. de artes	70919-970 Brasilia DF	EM von Rune Lindblad, Lucien Goethals, José Vicente Asuar, Jorge Antunes Conrado Silva, J. A. Mannis, Rodolfo Caesar, A. Kampela, Jorge Antunes
10. - 12.	Kurs „Technology and the Composers“ Uni. of Maryland; Dep. of Music	College Park USA, Maryland 20742	opening **concert** (Koenig, Nono, Cage, Erickson, Wilson, Dashow); concert II (Xenakis, Fuller, Bennett, Oliveros, Belet, Gibson, L. Moss); concert III (Risset, Yuasa, Laske, DeLio, Dusman). **Lectures** by : Laske, Yuasa, Risset
13.	Tage für Neue Musik; fax: +41-1-2121404	CH - 8022 Zürich	Experimentalst Strobel Stiftung: Luigi Nono (Diario polacco #2), K. Huber, H. Marti, D. Almada
15. - 18.	18. Tonmeistertagung **Deadline** für die Vorträge: 30.6.94 VDT; Masurenallee 8 - 14 D - 14 057 Berlin fax: 030 - 404 5927	Stadthalle Karlsruhe Organisationsbüro: Gisela Jungen Am Zaarshäuschen 9 51427 Bergisch Gladbach fax: 02204 - 21584	Schwerpunktthemen der **Vorträge**: Beschallung; Digitale Tonübertragung und ihre subjektive Qualität; Klangästhetik und Aufnahme-praxis; Mehrkanalton bei Film und Fernsehen; Mikrofone und Lautsprecher; Multimedia-Aspekte; Musik und Musikinstrumente; Psycho-akustik; Raumakustik und Raumwiedergabe; Regieanlagen und Übertragungstechnik; Studiomanagement; Technische Neuerungen und ihre Auswirkung auf den Beruf des Tonmeisters; Vernetzung von Produktion, Sendeablauf und Archivierung - Neue Speichermedien. **Produktforum**: Vorträge und Demonstrationen von Geräte-Neuentwicklungen **Industrieausstellung** (inzwischen eine der größten auf dem Gebiet der professionellen Studiotechnik)
17.	Jüdische Kulturtage; Akademie der Künste	Hanseatenweg, Berlin	Jeffrey Burns & Neue Pegnitzschäfer spielen: T. Murail, Levinas
18. - 11.	Kompakt-Workshop für max 15 Personen Kursgebühr erfragen Tel.: 0721 - 9340300	Ritterstr. 42 D 76 137 Karlsruhe	„Musikalische Arbeit mit Computern“, Leitung Johannes Goebel, Mitarbeit Dutilleux, Taube, Schweizer, Staff. Themenauswahl allgemein über „alles“ und Klangerzeugung, algorithmische Komposition, Raumprojektion. Vorträge, Diskussionen, Analysen.
24. - 26.	InselMusik / SFB Masurenallee	14 057 Berlin	Aufführungszyklus : Franz Martin Olbrisch (FMo99,5)
25. - 27.	Festival Neuer Musik, Fortbildungszentrum	21335 Lüneburg	Konzerte (mit H. W. Erdmann, Todd Reynolds), Aufführung von Preisträgern „Bourges“, Workshops

Dezember 1994

1.	De Ijsbreker fax: +31-20-6946607	Weesperzijde 23 NL - 1091 EC Amsterdam	EM aus Kanada von u.a. Serge Acuri, Jean Piche und von Paul Steenhuizen, Allison Cameron
2.	Jour fix 20 Uhr; Akademie der Künste	10117 Berlin	offene Gesprächsrunde im Akademiestudio unter Leitung von Georg Katzer
2. - 4.	Kurs SPNM / Sonic arts	GB - Uni. of Bath	Introduction to state-of-the-art computer sound synthesis. Call for Pieces **Deadline** 31.10.94
5., 7.		Kyoto und Tokio; Japan	Instrumental - elektronisch - audiovisuelle Werke von Wilfried Jentzsch
7.	ZKM in der Fabrik	Lorenzstr. Karlsruhe	Erhard Grosskopf (Hell 1)
9.	De Ijsbreker fax: +31-20-6946607	Weesperzijde 23 NL - 1091 EC Amsterdam	Tera de Marez Oyens spielt eigene Werke u.a. mit EM
	Hebbeltheater	10963 Berlin	Hans Peter Kuhn & David Moss: Wild Wires Tactile Tales
31. **Deadline**	competition CDMC & RDE	CDMC, Santa Isabel 52 E - 28012 Madrid	Wettbewerb Radiophonische Musik (für das Radio, Sprache beliebig) unabhängig von Nationalität und Alter. Zugelassen sind bis zu 6 zusätzliche Interpreten. Die Einsendung von Projektbeschreibung plus Klangbeispielen reichen dann aus, wenn das Werk noch nicht produziert ist.

1995

2.2.	De Ijsbreker Weesperzijde 23	NL - 1091 EC Amsterdam	Het Elektrisch Laagland, EM des CEM-Komponisten-Kollektievs und aus dem Archiv des CEM Utrecht
18., 19.2.	15th Birthday celebration sonic arts network	South Bank London	composer´s forum, percussion concert, international Computer music concert, Tim Brady - concert
20.2 - 5.3.	Geographie der Wahrnehmung; BMUK 0043 2732 - 82669	A - Krems / Minoritenkirche	Komposition, Cybersound, Phämomenologie: Maryanne Amacher, Georg Friedrich Haas, James Tenney; mit ensemble recherche u.a. …
2. - 5.3.	5th Biennale & Symposium on Arts and Technology at Connecticut College	USA New London, CT 06320	email: cat@conncoll.edu
10. -13.3.	Oper Leipzig postfach 346	D - 04003 Leipzig	Karlheinz Stockhausen: FREITAG aus LICHT. Uraufführung. 20-kanalige Tonprojektion der WDR-Produktion
19.3	Royal College of Music	GB London SW7	J. Hyde, T. Howle e.a.
31.3 **Deadline**	Weltmusiktage 1996 Danish Music Information Center	Grabrodretorv 16 DK- 1154 Kopenhagen K	Die Voraussetzungen sind die gleichen wie immer (siehe Kalender). Beachte Kalender letzter Eintrag 1996 ! fax: +45 - 33 - 322016
27.4.	De Ijsbreker; Weesperzijde 23	NL - 1091 EC Amsterdam	EM aus den 50er Jahren von Nono, Berio, Maderna, Evangelisti, Varèse. fax: +31-20-6946607
19.5.	De Ijsbreker; Weesperzijde 23	NL - 1091 EC Amsterdam	mikrotonale Musik: J. Mitchel & Ph. Mead spielen: Pascal Criton, Allan Schindler, Rodrigo Velloso, Horatiu Radulescu
ab 31.5.	ASA	USA Washington DC	Conference of the Acoustical Society of America
7. - 10.6.	KlangArt 95; Musik & Elektronik fax: 0541 - 42213	Heger-Tor-Wall 19 49074 Osnabrück	Thema „Musik und Elektronik“. 3 Veranstaltungsbereiche KonGress, MusiTec, Festival. Konzerte, Vorträge und Workshops für alle, die sich umfassend über den kreativen Einsatz der neuen Technologien informieren wollen.
22.6.	De Ijsbreker; Weesperzijde 23	NL - 1091 EC Amsterdam	Karlheinz Stockhausen (u.a. Gesang der Jünglinge)
23.6. - 2.7.	Weltmusiktage 1995 GNMR Tel.: 0201 - 24525 0 Fax: 0201 - 233257	Postfach 104242 D 45042 Essen	Motto: an der Schwelle zum 21. Jahrhundert. Ausschreibungs-Kategorien: A. Orchesterwerke B. Kammermusikwerke C. Werke für Kammermusik (alle 3 mit oder ohne Elektronik); D. Performance / Multimedis / Experimentelles. **E. Elektronische Werke**. Internationale Jury (Wolfgang Becker, Toshio Hosokawa, Nicolaus A. Huber, Zoltan Jeney, Graciela Paraskevaidis, L. J. Tamusuza, Christian Wolff) **suchte aus**: Kategorie **A**: Kessler (Aufbruch), Kategorie **C** : H. Nakamura (Drawing), Mesias Maiguashca (Moments musicaux), Katharine Norman (Icarus). Kategorie **D** : Hans Otte (Räume), Urs Rickenbacher (Klangzonen); Kategorie **E**. (EM): Gerald Eckert (Aux mains de l‘espace), Lee Jung Hae (Yun Mu 3), Gilles Gobeil (La où vont les nuages), Pete Stollery (Shortstuff), Ronald Alford (Image: the pop can…)
2. - 6.7.	ISMA 95; IRCAM	1 place I. Stravinski; Paris	International Symposium on Musical Acoustics in Dourdan (near Paris). contact: email isma@ircam.fr
4. - 10.9.	Gaudeamus Music Week Gaudeamus Foundation	Swammerdamstraat 38 NL 1091 RV Amsterdam	keine EM für Wettbewerb (Deadline 31.1.1995). Aber Vorstellung preisgekrönter Werke von z.B: Bourges
	ICMC‘95; Banff Centre for the Arts	Banff / Canada	
Okt.	ISEA‘95	Montreal / Canada	
1. -3.11.	INTERFACE 3; Kulturbehörde Hamburg Prof. Dr. Klaus Dencker	Hamburger Str. 45 D - 22 083 Hamburg	Motto „NetzWelt“. Themen: Netze, Modelle, Ethik und Ästhetik, Schnittstellen, Gefahren. 30.4.94 (Deadline)

1996

August	WFAE	France (?)	international meeting in France of the World Forum for Acoustic Ecology (focus „the sounds of transportation“)
7. - 14.9.	Weltmusiktage 1996 Danish Music Information Center	Grabrodretorv 16 DK- 1154 Kopenhagen K fax: +45 - 33 - 322016	Schwerpunkte: Musiktheater und **interactive computer music**. Wichtige Kategorien: **„electroacoustic works“** (multimedia, interactive computer works, tape works), **„installations based on music“** (Museum of modern art, autdoors and indoors in the period around the festival)