# Cross-Art Collaboration in New-Technologies

Brian Curson

2004

# Towards An Understanding of 3D Virtual Environment Software Development and the Choreographic Process

This essay was written as part of an MA module in Collaborative Arts at University College Chichester, UK and presented on 28th June 2004

# Table of Contents

# 1 Introduction

Since 2000 Robyn Stuart and I have collaborated, as al'Ka-mie, in research into 3D Virtual Environments (3DVEs). These 3DVEs are created by projecting a view of a computer 3D model, as seen by a notional virtual camera, onto a white stage space composed of white floor, cyc and screens. Onstage performers may appear to inhabit the model. If the virtual camera is moved through the computer model then the audience can have the feeling of moving physically through the virtual world.

In the last year, as part of Robyn's MA research at UCC our research has focussed on investigating how this virtual environment movement, which we call the virtual choreography (Vc), interacts with corporeal movement onstage, the physical choreography (Pc).

I contributed to this research by developing software that could control the movements of the virtual camera, and also by using my dance background to operate the Vc system. I wrote software[1] that could control the movement of the virtual camera in real-time using two joysticks. During this year of intense research I made numerous developments to the software. In a recent research intensive 'Collaborations in New Technologies' at UCC we worked with three dancers[2]. This intensive provided an opportunity to develop our software further, and allowed me to

[1] This software was written in Lingo, the programming language of Macromedia Director
[2] Guy Adams, Becky Yates and Ayesha Mill

undertake a major study reflecting on the two-way relationship between the system operator (his choices and aesthetics) and the functional development of the software. I discuss this relationship in this essay and suggest ideas for future development, future development strategies and areas of future research.

# 2 Theatrical Space as the Computers Body

> In an analogy to the human body, the theatrical space is the computer's body, the electronic media the limbs, cameras and microphones used as sensors are the eyes and ears, a speech generation program the mouth, and the CPU(s) and internal programming are the brains that are used to interact with the physical world. The space that holds the performance becomes an environment generated from behaviors of the computer, responding to and shaped by performers, designers, and technicians.
> (Lovell, 2000)

Lovell (2000) continues by recognising three abilities that computer intelligence in the interactive arts must have: *Perception*, *Reasoning*, and *Dexterity*. Using this as a general model for intelligence I suggest that a real dancer can be represented as in diagram 1. Dancers simultaneously *perceive* their surroundings, the actions and locations of other dancers, the forces exerted on and by their bodies and the effect of sound and music. This input is fed through their *reasoning* to their *dexterity* resulting in articulated movement of their bodies.

## 2.1 Introducing The Virtual Dancer

Extrapolating Lovell's (2000) analogy further, in relation to al'Ka-mie's work, the *projection from the computer* would become the computer body and *the movements* *of the computer model* would be the dance of the computer's body. However, Lovell's (*ibid.*) model connects the computer directly to sensor equipment allowing it a direct perception of the performance space. This does not occur in our research to date, rather a computer operator (typically myself) perceives the performance space and interprets actions therein to give input to the computer, which in turn interprets this input to give an output in the form of the movement of a 3D model (Diagram 2). The syntheses of the computer operator and the computer form a virtual performance entity, capable of perceiving events onstage and using its total reasoning power to articulate the movement of the 3D model, the computer's body. This entity can be treated as a virtual dancer, just as Lovell (*ibid.*) suggests a computer operating theatrical effects in a performance can be treated as a virtual performer. The use of this perspective may inform our understanding of the functional processes and development of this virtual dancer. Further our general knowledge of ourselves may grow, as Heim suggests:

> While recognising the computer as a component in our knowledge process, we can attend to what happens to us as we collaborate with technology. Because human history is a path of self-awareness, as we deepen our understanding of computer interaction, we will also increase our self-understanding. (Heim, 1993, p70)

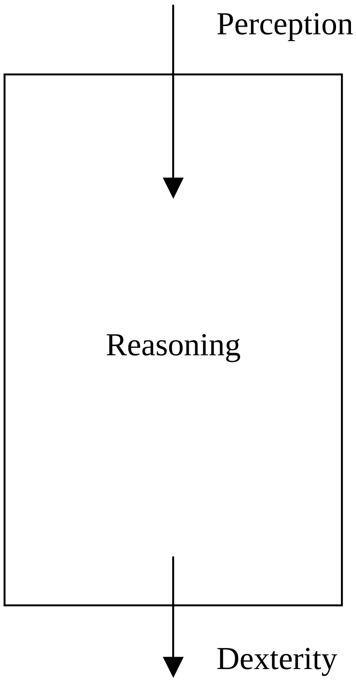


**Diagram 1 –** The Real Dancer

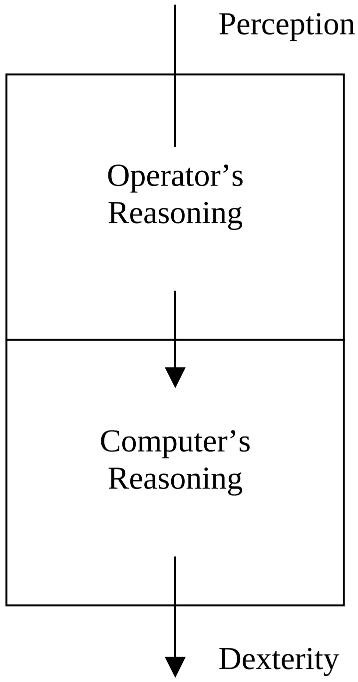


**Diagram 2 –** The Virtual Dancer

## 2.2 The Perfect Improviser Dancer - A Notional Model

In a notional model of an expert dancer in improvisation, this dancer would have basic skills of remembering sequences and editing them, for example by changing dynamic or adding punctuation. They would be capable of creating material de nouveau or selecting from a palette of set material to transmogrify it in response to real-time events, e.g. sound and the presence and tactility of other dancers. This dancer would be capable of remembering intricate scores and adapting their movements to fit these scores. They would also be aware of the choreographic and narrative integrity of the whole piece as it unfolds and have a sense of its emotional resonances and be able to make further improvisational choices based on this understanding (see Blom and Chaplin, 1982). Finally they perform with the constraint of physical continuity, that is, one movement *must* link into the next and they *must* work within the parameters of their body facility.

# 3 Current Chronological Software Development

| # | Function | Description |
|---|---|---|
| 1 | Record and Playback | This function allowed us to record model movements as a sequence of positions called an SEQ. |
| 2 | Timeline | SEQ recorded direct to a timeline structure allowing multiple SEQs to be combined & edited. |
| 3 | Music Playback | Allowed music files to be added to the timeline and played when the projection was playing. |
| 4 | Relative SEQs | Ability to change SEQs from 'explicit' to 'relative'. An explicit SEQ defines not only the movement of the model but the position in space. A relative SEQ just defines the movement. |
| 5 | Multiple Models | A model palette was added. SEQ now would not only record the movement within the model but which model it was recorded in. Playing sequential SEQs recorded in different models would have the effect of cutting from model to model, just as a film cuts from scene to scene. |
| 6 | Palette Pages And synchronisation | Ten palette pages were created, each page containing 29 SQP members arranged as a keyboard, each member corresponding to a specific key. The members could contain a timeline with a number of SEQs, thus becoming a Vc phrase. When the member's identity key is pressed the relevant timeline plays. The SQP palettes gave the ability to organise Vc phrases into palette groups (structure).<br><br>Palettes gave synchronisation: A Vc phrase was constructed to augment and complement a Pc phrase. This Vc phrase was broken into small sections, each assigned to a SQP member in the same palette. These sections could be cued by Pc events on stage, and thus the Vc could be made to follow the Pc. This gave a crude dialogue between the computer dancer and a real dancer. |
| 7 | Retrograde SEQs | This allowed us to take a specific 3DVE position, define a movement from it and then retrograde it giving us a movement that culminates with that specific 3DVE position. This is useful when a Vc movement needs to finish with a particular strong defined visual composition. |

**Table 1** - Software functions added (chronological)

In Table 1 I have depicted the chronological development of al'Ka-mie's software. Similarities can be drawn between the functionality of the developing software and the model of the notional expert dancer described above. The SEQs are akin to the dancer's movement sequences; the timeline is akin to the dancer's ability to edit and combine sequences; the SQP palettes are akin to the movement palette from which the real dancer works; and a sequence of SQPs (a SQP run) is akin to the ability of a dancer to dance and interact with other dancers.

## 4 The Developmental Process

As I was both the computer operator and the programmer of the system in al'Ka-mie's work, these functions were closely integrated and this integration significantly affected the creative work and developmental processes of the software. In discussing this, I refer to the operator/programmer in the third person, as Brian, and consider the developmental changes which occurred, as part of Lovell's second computer intelligence ability, that of *reasoning*.

The first program Brian created allowed the computer to handle only basic movement control of the model, via joysticks, while functions such as movement memory were Brian's sole responsibility. However, accurate manual recreation of joystick movement was difficult to achieve. Thus Brian programmed the computer to record the joystick movements and consequently transfer the movement memory function to the computer. Now, Brian could stop and start the recording, and manually edit the virtual movement.. Brian subsequently also transferred this editing function to the computer by creating timeline editing. This cyclic developmental process was repeated progressively adding more complex functionality. See Diagram 3. This process of function transferral over time built a complex layered decision-making function in the computer, compared to the original basic in-real-time joysticks control.

However, this cyclic developmental process was not just function transferral from Brian to the computer. It was also a process of the computer enabling Brian to codify and increase the functionality of the virtual dancer (that is the combined entity of Brian and computer as performer). I see this as a kind of evolutionary symbiosis occurring between human and machine, both being dependent on the other for their evolutionary development. This relationship demonstrates the human-cyborg development as discussed by Virtual Reality theorists such as Biocca (1997) Ihde (2002) and Heim (1993) who notes that:

> Even the research and development at major corporations is now moving away from artificial-intelligence research, in which the computer functions separately, to research on the human-computer symbiosis, including information environments that augment human bodily perception and create "virtual realities". (pp. 60-61)

Investigating this relationship further, I find that Brian is making a choice of which functions are transferred to the computer. These choices may appear to be simply increasing the complexity of operations conducted by the computer. However, exactly how the software is layered and organised and the boundaries of user operation are expressions of Brian's aesthetic. For example, Brian's aesthetic is for close movement synchronisation between the Vc and Pc and this aesthetic has led to his developing many SQPs within many palette pages (see Table 1), allowing each small movement of a dancer to be closely followed by a similar small Vc movement. If Brian did not have this aesthetic, it is likely he would have sectioned and organised the Vc quite differently, perhaps by movement function rather than relationship to dancer's Pc. Thus, this developmental sequence is not only the 'synthesis of a cyborg', but also the formulisation and codification of Brian's aesthetic.

The UCC intensive represented the most recent recursion of the developmental cycle – and lead to the idea of the codified Score.

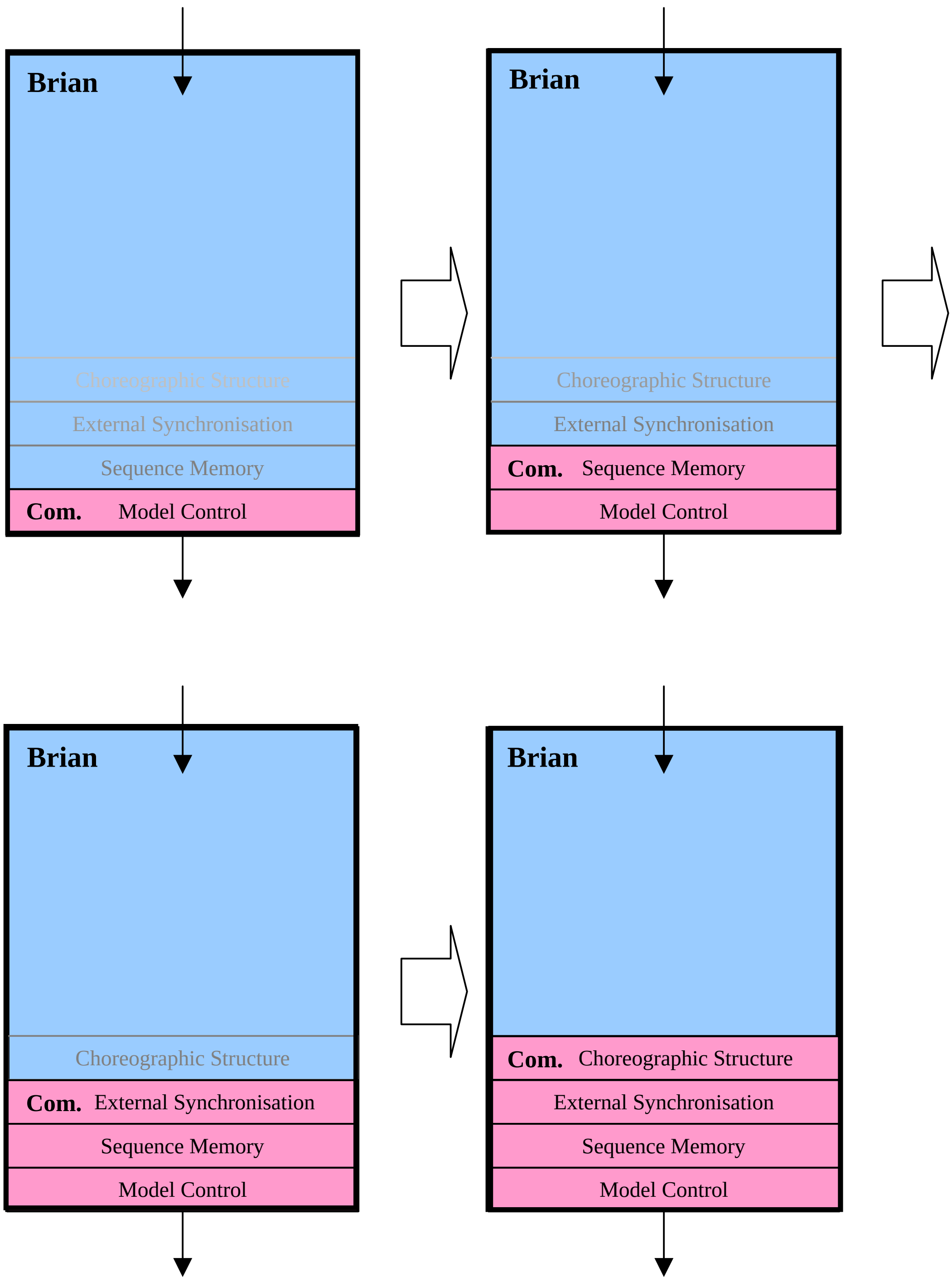


**Diagram 3** – Software Developmental Process Showing Transference of Functionality from Operator/Programmer (Brian) to the Computer

# 5 *FLUX* – An Experiment in Scores and Hypertext

## 5.1 Experimental Method

At the UCC intensive we worked with three dancers, Guy, Becky, and Ayesha, to create a short research piece, (*FLUX*) which investigated the use of hypertext in our work (Landow, 1997). Each dancer was guided into creating a Pc phrase specific for a particular VE model. Some dancers were asked to learn others' phrases. All dancers knew the Graveyard phrase. Each phrase had a set movement order but was not set temporally, i.e. the dancer was free to change dynamic. A Vc phrase was created to complement each dancer's Pc phrase. Each Vc phrase was placed in one palette, but in divided sections i.e. several SQPs (see Table 1). By stepping through the palettes, and cueing the Vc sections to Pc events, the Vc phrase could be made to follow the Pc phrase (as described in Table 1). The palettes are represented in Table 2.

| Dancer / Model | Palette Entries | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | B1 | B2 | B3 | B4 | B5 | B6 | B7 | | |
| Guy / Maze | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G9 |
| Ayesha / Temple | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | |
| Graveyard | GY1 | GY2 | GY3 | GY4 | GY5 | GY6 | | | |

Table 2 – Layout of Vc Palettes for Hypertext Experiment

FLUX's score consisted of seven sections (Table 3). If during the score dancers completed a Pc phrase they were instructed to start again at the beginning. The score pathway though the Vc palettes is represented as the orange arrow in table 2. This pathway was dynamic and varied between performances depending sometimes upon the Vc operator's decision and sometimes upon the dancers' decisions on when to change to the next section, for example 'whether Guy had started the Graveyard Phrase'.

| Section | Description |
|---|---|
| Introduction | Becky onstage, Geometric Shapes model zooms into view. |
| Section 1 | Becky starts performing her phrase in the Geometric Shapes model. Brian steps the Vc through her palette B1…B7, synchronising with her Pc phrase. |
| Section 2 | The Brian changes the Vc to the start of Guy's Vc, G1, in the Maze. Guy enters and performs his phrase. Brian steps through the Vc in his palette, G1…G9, Becky remains still wherever she was when the model changed to the Maze. |
| Section 3 | Brian changes back to the Geometric Shapes model, Becky continues where she left off and is joined by Guy performing her Pc in synchronisation with her. Brian continues to step the Vc through Becky's palette. |
| Section 4 | Brian changes back to the Maze, Guy continues where he left off, Becky is still. Brian steps the Vc through Guy's palette continuing from Section 2. |
| Section 5 | Brian changes the model to the Temple, cueing Ayesha's first Vc section A1. Ayesha enters and starts performing her Pc phrase. Guy joins Becky being still. 30 seconds into Ayesha's phrase Becky joins Ayesha in a synchronised duet. Brian steps the Vc through Ayesha's palette, A1…A8. |
| Section 6 | Brian changes to the Graveyard. Guy starts performing the Graveyard phrase, both Becky and Ayesha join him. Brian changes the Vc to GY1 and steps through the Graveyard sequence GY1…GY6 following the trio as they perform. |
| Section 7 | Brian changes out of the Graveyard model. All of the dancers are now free to perform their own sequences or to follow one of the others that they know according to some structured rules. Brian is free to switch the Vc to follow any of the three dancer sequences. B1…B7, G1…G9 or A1…A8. Brian can choose to terminate the work at any time by fading lights and sound. |

**Table 3 –** Overall Score for Hypertext Experiment

## 5.2 The Generic Score Structure

In the following I suggest a way to take the score rules that I was following as an operator or could follow as a dancer to define a structure that the computer could use to follow any score, a generic structure.

During the performance of an improvisation score a dancer continually monitors the environment for a small set of conditions. When a specific condition arises they alter their Pc accordingly and perform the next specific task. Returning to the model of computer/operator as virtual dancer (Section 3.0), I suggest the generic score structure as a set of *statements*, with each *statement* performing a particular task. This task is continued until a condition (from a list of conditions) is met. This condition then directs the computer to another statement within the score or to END, the end of the score. The general structure of the statement is illustrated in Diagram 3. Not only can the computer/operator entity be viewed as a *dancer*, but also it can follow a similar set of operating instruction as for an improvising dancer.

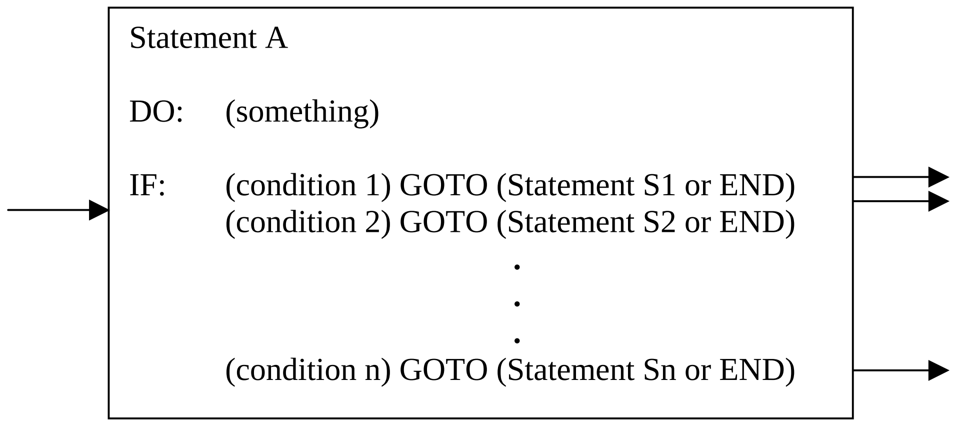


**Diagram 3 –** The general structure of a score statement

## 5.3 *FLUX* expressed in the Generic Score Structure - Nested Scores

*FLUX* described above can be thought of as a number of sequential scores (the dancers' phrases) within a multi-branched main-score (the whole piece). For each dancer's Pc phrase there is a sequential score controlling and synchronising the Vc to the Pc. A section of Guy's Vc score is illustrated below in Diagram 4.

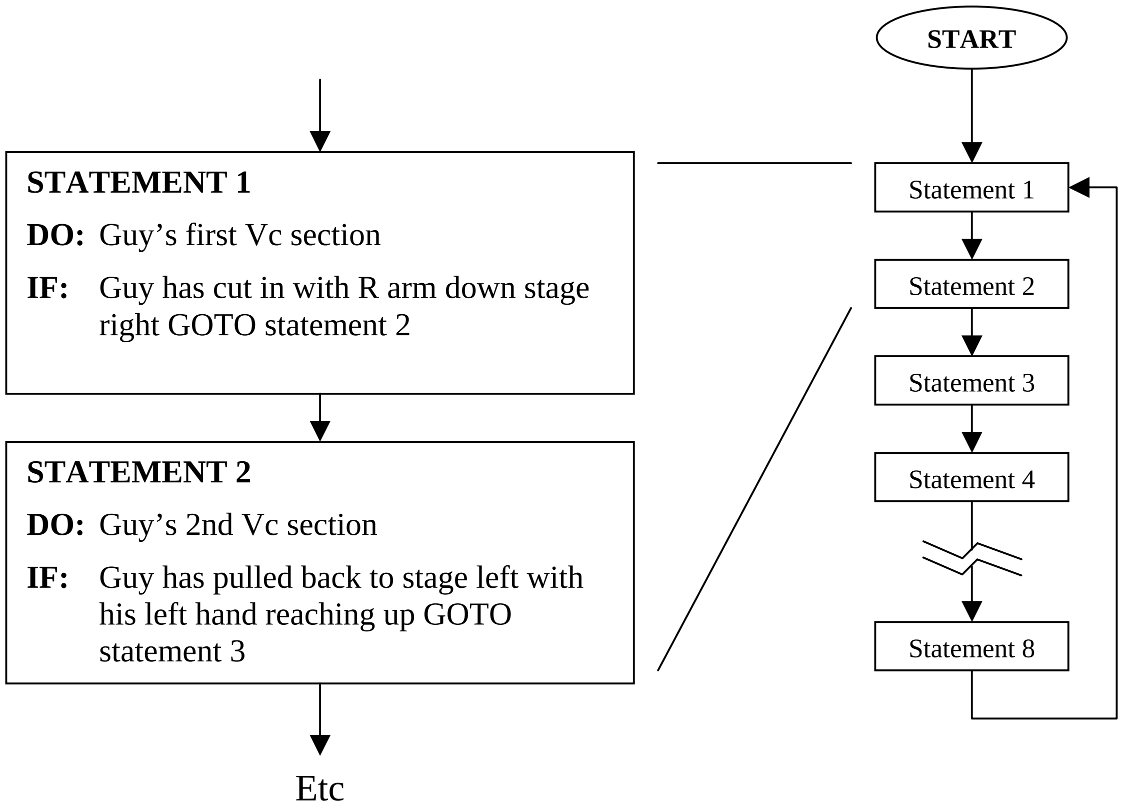


**Diagram 4** – Guy's Vc Sub-score

Thus the dancers' Vc phrases are indefinite sequential scores (without a specific end) consisting of statements that execute each Vc section in turn while testing for the cue for the following section.

**Diagram 5**
– FLUX's Main-Score

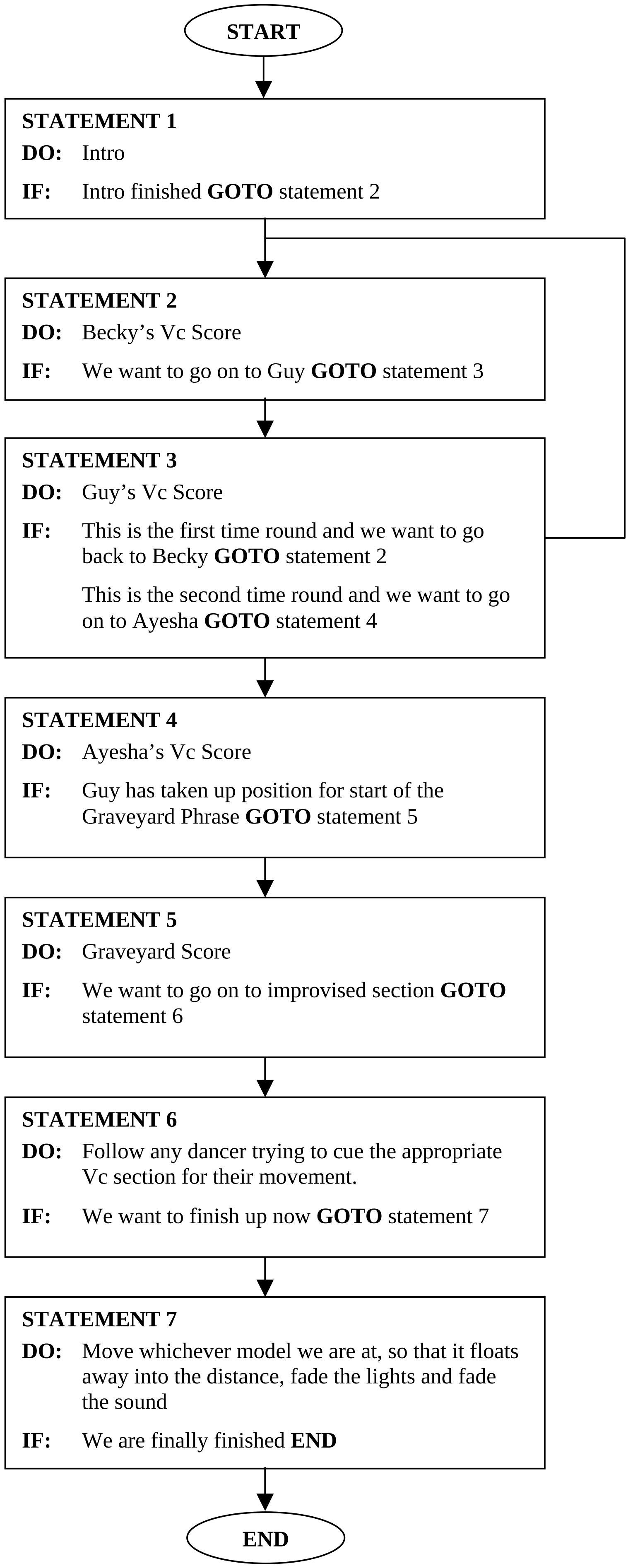

Diagram 5 shows the full experimental score transposed into the standard format. It contains four 'nested' 'sub-scores'. Note that when a sub-score is reactivated as in the second execution of statement 2, the sub-score 'Becky's Vc Score' would continue from wherever it finished the first time around. Also the set of conditions of a main-score-statement still remains effective during the execution of a sub-score by that statement. Thus as we progress through Becky's Vc score, there continues to be a choice of progressing at any time onto statement 3 and Guy's score. As the score progresses the operator will be presented with an ever-changing set of pertinent conditions to test for or choices to make.

## 6 Structural analysis – Maximal Combinability

The structure that arose out of the software's developmental process (analysed in Section 5) is predominantly hierarchical. In the software, SQP palettes contain timelines that contain SEQs.

In this model of a dancer the body is controlled by movement sequences that are in turn controlled by timelines. This is a simplistic understanding in which the dancer has been reduced to a perfect playback device, a VCR of movement. In our notional model of an improvising dancer they are capable of taking many forms of impetus such as image, music, emotion and proprioception and combining them fluidly to inform their movement. This functionality[3] of our notional dancer forms a more sophisticated web-like structure.

[3] Here I use functionality to refer to wide range of capabilities of the dancer including imagination, proprioception and musicality

In this model the various functions of the dancer are endowed with equal importance and operate more as a team of specialists working together for creative resolution. Diagram 7 depicts the potential interplay of various functions of a dancer. Some of the connections will develop, through use, to be stronger, while others might be less used, possibly contradictory and become weak. Through differential use different types of network could develop between these functions, but this network can over time fluidly change it's structure. Likewise, it is important that the functions that the computer software handles are capable of being combined in a wide fluid variety of ways. I call this the principle of *maximal combinability*. This versatility makes a computer programme more powerful.

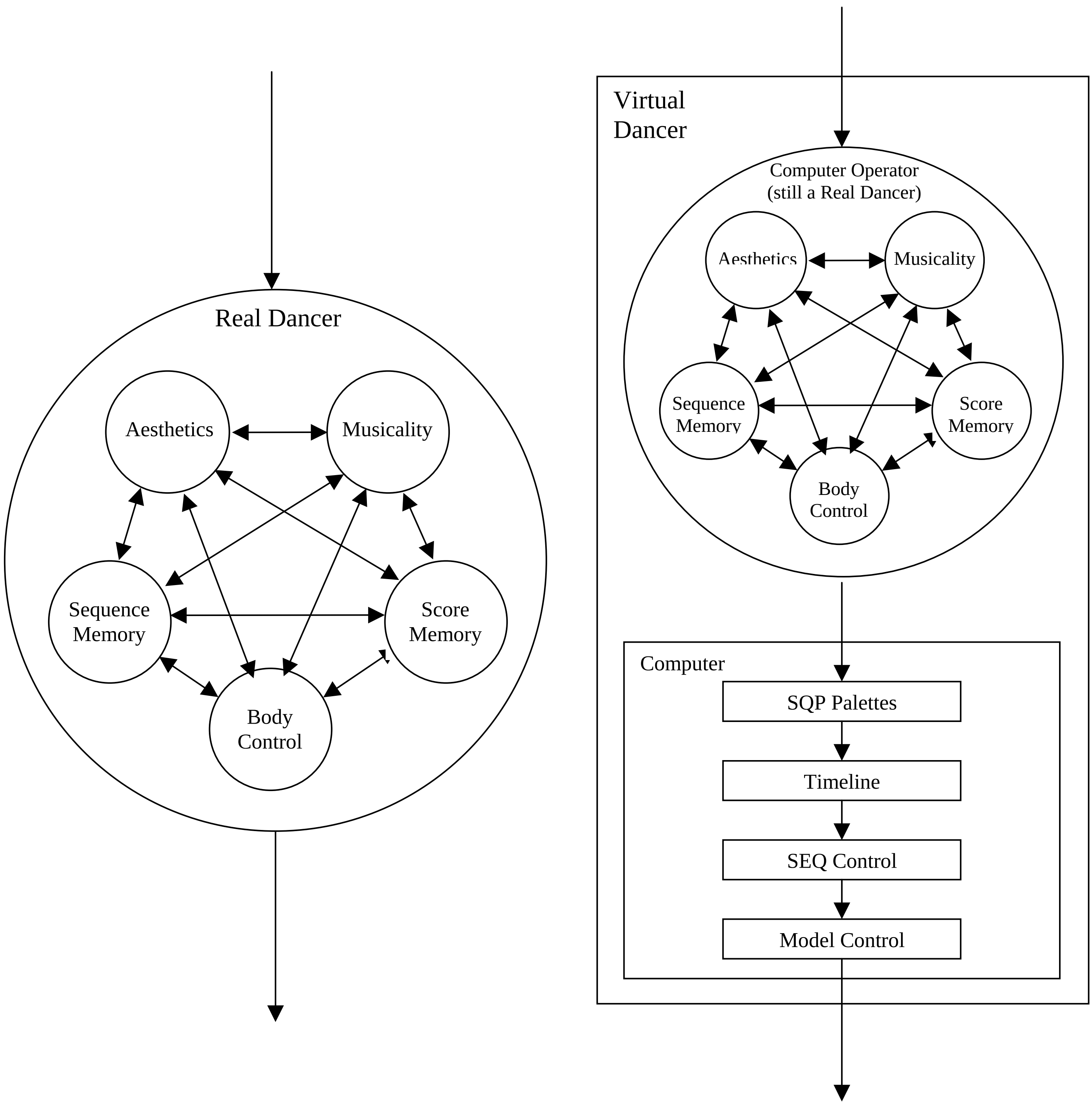


**Diagram 7 –** Free interplay of functionality of a real dancer and of the computer operator vs. hierarchical functionality of the computer

# 7 Space and Time and Putting Them Together – The Elements of Dance

Few introductory books into choreography (e.g. Blom and Chaplin, 1982) will neglect to mention the two fundamental elements of dance, space and time. In this section I use these ideas to inform my software development.

## 7.1 Space

The virtual dancer's physical expression within space is the projection of the 3DVE model (see section 2). The computer processes that control the 3DVE model constitute the computer's 'understanding' of space, which in al'Ka-mie's current software is the SEQ. The basic SEQs represent a simple list of camera positions in time.

The relative SEQ (see table 1) however represents a divergence from defined material to defined process. Instead of saying move back 3 steps from point A to point B it says move back 3 step from *wherever you've got* to. This process-based understanding of space is more akin to how a real dancer would work. Extending this analogy we could develop processes such as; take 10 seconds to move to point C passing through points A and B on the way, move in a wobbly fashion around where you are or move in a straight line from wherever you are to point A. With the right formula a process could also imbue the Vc with a sense of inertia mimicking the motion of the physical bodies to which it may relate. These processes I refer to as *Space-Functions*.

### 7.2 Time, The Timeline and the Score

A timeline and a score would seem very different entities, one dealing with the continuity of time, while the other deals with conditional progress or probability. I see them as the two extremes of the same continuum. The execution of a timeline continues on the condition it has not finished, when it reaches the end it stops, while without time within which to do a thing and in which to test for conditions the score statement would be meaningless. In the simple single statement score 'Do something IF it hasn't finished' we see both a timeline and a score. We could also develop the idea of the timeline to include a number of channels that could be enabled or disabled as the timeline runs. Thus the timeline, like the score, could support conditional branching..

### 7.3 Combining Space and Time

Using the principle of *maximal combinability* explained in section 6 and the elements of Timelines, Scores and *Space-Functions* (section 7.1), I envision software that could combine these elements in the widest number of ways, allowing multiple recursions. Timelines could contain any number of channels, containing any mix of sub-Timelines, Scores, and Space-functions. Scores could contain any number of statements, each containing a sub-Score, Timeline or Space Function.

In constructing the combinations of timeline and score we are effectively exploring the continuum between pure timeline and pure score. Some of these combinations will probably be mutually inconsistent but it is this possibility to create the absurd that also gives us the possibility to create the sublime.

### 7.4 Combining Space, Timeline and Scores – An Example

Diagram 8 illustrates a timeline containing a score and a timeline. As the play head progresses along the top, the timeline plays. The operator is presented with two choices: to enable the *FLUX* score (see section 5.3) in channel 2, (Choice 1) or to enable Timeline 2 in channel 3 (Choice 2). If (s)he chooses Choice 1 then the score will play when the play head enters it. At any time during the execution of the timeline (s)he may choose Choice 2, suspending the score and enabling Timeline 2. If the play head in timeline 1 is half way into the occurrence of timeline 2 when Choice 2 is chosen then timeline 2 plays from half way. Choice 3 is now added to the choices that the operator may choose. If the play head of Timeline 2 is in the Music Vc Timeline (containing a Vc phrase constructed to augment the Music track in the Timeline 1) this Vc phrase plays from that part of the timeline. If now the operator chooses choice 3 and the play head of Timeline 2 is in the Space-Process ‘Wobble’ then the model starts wobbling. At anytime the operator is free to choose Choice 1 again and resume the *FLUX* score from where it left off. The Music Vc Timeline would then cease playing as would the ‘Wobble’ Space-process.

Effectively we have created a means for the operator to switch between a Vc - Pc relationship in the form of the *FLUX* score and a Vc - Music relationship (with an added bit of wobble effect at a specific point). The Virtual dancer would be capable of choosing to relate to the music or to the other dancers

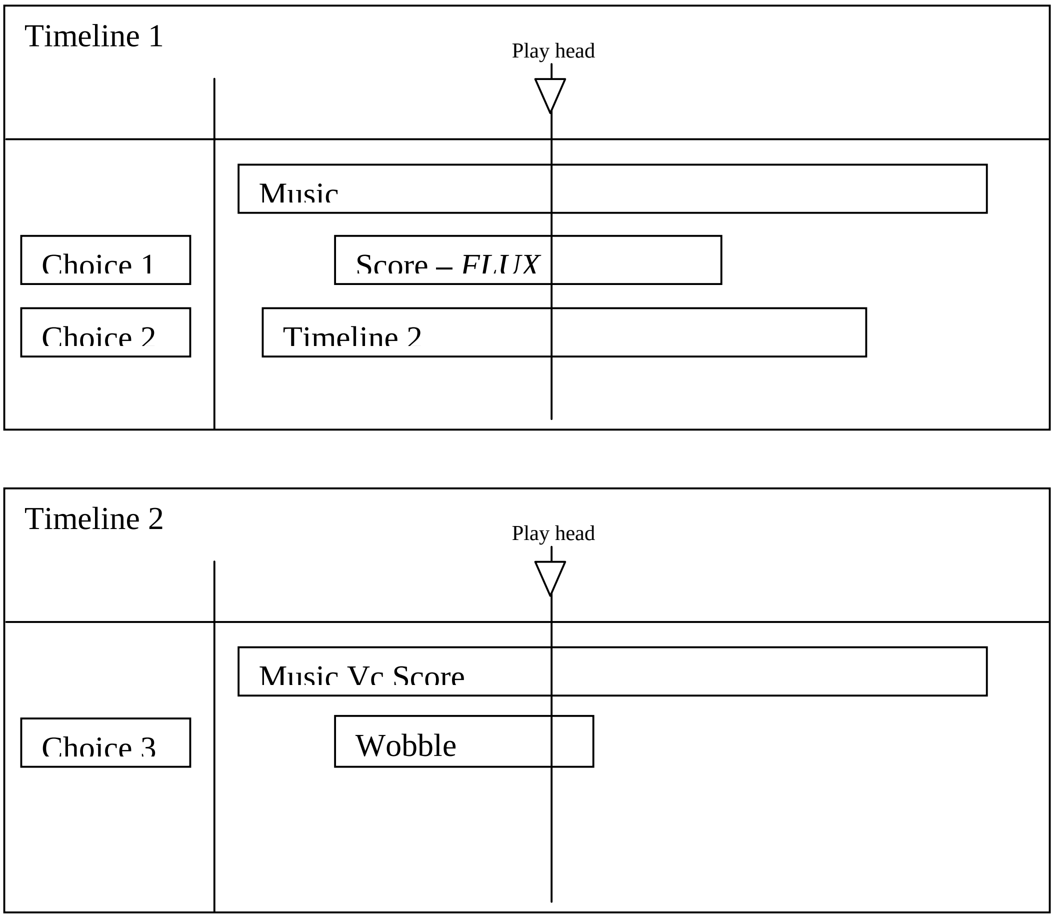


**Diagram 8 –** A timeline containing score and sub-timeline and the ability to choose between them.

# 8 Other Score Structures and Narratives Resonances

Different score structures could lead to different narrative resonances. (Some sample structures that scores could form are shown in Diagram 6.) In the experiment *FLUX* the iteration of Becky and then Guy's Vc scores sets up a relationship between them with an asymmetry of Guy joining Becky on her second iteration. The switching then to Ayesha interposes on this relationship; Becky joining Ayesha strengthens this interposition. The multi-branched main-score of *FLUX* sets up a specific narrative resonance.

Film uses complex score structures (Bordwell, 1997, 1985). Movies can be thought of as sections containing scenes containing shots, with the shots from a specific scene forming a sub-score. Cinematic techniques such as parallel cutting, where the film cuts from one scene to another and then back to the first to give the effect that both scenes are occurring simultaneously are examples of different types of score construct. Such visual techniques can give rise to a hypertext structure (Gaggi, 1997) and can infuse a Deleuzian Time-Image (1989).

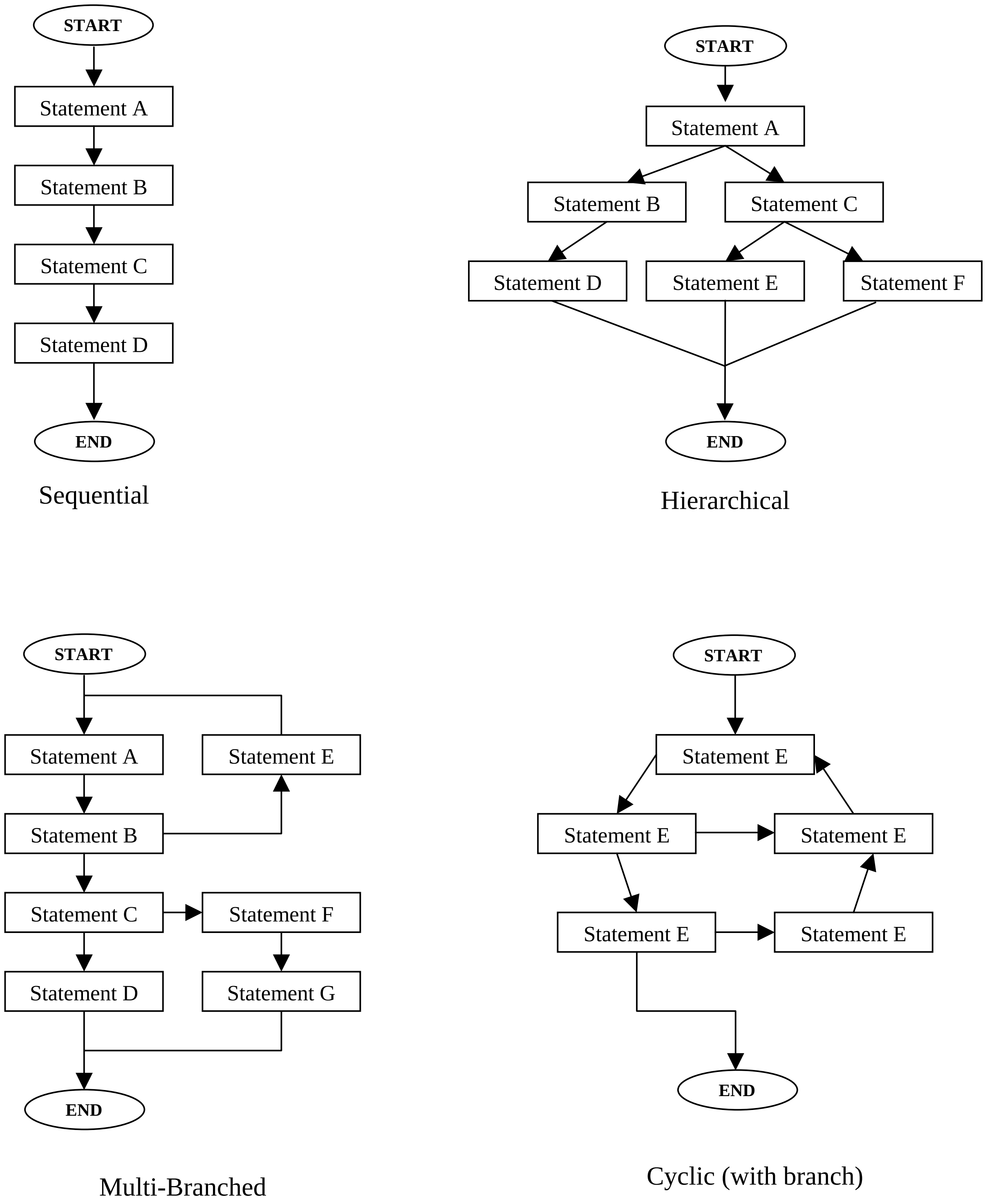


**Diagram 6** – Examples of score structures

# 9 Future Areas of Research, Limitations & Conclusion

In this essay I have used my understanding of the processes intrinsic to a dancer to reflect upon the history and potential future development of al'Ka-mie's software. The consequence of this reflection was a fundamental shift of thinking from an engineered hierarchical model of the software to a networked (potentially rhizomatic see Deleuze and Guattari, 1987; see also Stuart 2004) choreographic model of the software.

However despite this change of thinking, the computer would remain no more than a machine providing the operator with pertinent choices at the correct times from a fixed set of rules i.e. algorithmic knowledge (Lovell, 2000). Could the machine learn, from rehearsal to rehearsal as a dancer does and maybe even from operator to operator as a dancer learns from different choreographers?

The computer is reliant on the operator/programmer for both it's perception of the performance space and it's development. The virtual dancer is separated into operator and computer. This represents a cleft in its collective reasoning bridged only by mouse clicks and key presses during execution of the score/timeline. This leads to a division of labour between the two where the operator perceives the performance space, drawing inferences to test the simple rule-based Boolean conditions that the machine presents him/her. However the use of the two joysticks, that are currently used only to control virtual environment movements and provide a multi-channel analogue input, could make this operator/computer interface more eloquent. For example the joysticks could be used to track the position of a dancer in the performance space. The fluid reassignment of the joysticks to various functions within the system is another example of *maximal combinability* (see Section 6).

Further, could the machine, given a direct form of sense of the performance space, draw inferences of it's own and reform it's own rule sets? Could the machine present alternative ideas to it's operator, adapting to whether those choices are chosen or not, as a dancer adapts to different choreographers, learning the type of material best to present them? The area of artificial intelligence is another major area of research for this project.

Finally, I have made much of the analogy between the virtual dancer and a real dancer. By virtue of their physicality a real dancer's body must be continuous in time and space. They cannot disappear in one part of the stage to reappear instantaneously in another, whereas the Vc can do precisely this. Furthermore, the Vc can move instantaneously from one computer model to another creating a cinematic cut from scene to scene. The impact of cinematic film theory on the software is a whole area of research that bears further investigation.

Lovell (2000) says that in theatre the computer will never replace the human, he continues:

> But the computer will give people more power to integrate media into productions, create interactive effects, coordinate events, and will provide more expressive capabilities to performers, directors and designers, empowering and enabling the creation of new forms and theatre events. **Lovell (2000)**

# Cross-Art Collaboration in New Technologies

Brian Curson 2004

**Student ID: 0306447**

# Towards An Understanding of 3D Virtual Environment Software Development and the Choreographic Process

## Bibliography

# Bibliography

.

Adventures in Motion Pictures (1996) *'Swan Lake'* choreographer Mathew Bourne, London; Video produced by BBC

al'Ka-mie (2002) *Rivan* performed at Richmond's Leap into Dance Festival' (London) and The Rondo Theatre in Bath.

al'Ka-mie (2001) *NT_001454* performed in'Resolutions! 2001 (London), Centredance Performance Night in 'Richmond's Leap into Dance Festival' (London), 'Northern Exposure' at The Bull Art Centre (London) and (2002) The Rondo Theatre (Bath).

Blom, L. and Chaplin, L. (1982) *The Intimate Act of Choreograph*y, London: Dance Books

Biocca, F. (1997) 'The Cyborg's Dilemma: Progressive Embodiment in Virtual Environments.' *Journal of Computer-Mediated Communication* [online] 3 (2). Available from: http://www.ascusc.org/jcmc/vol3/issue2/biocca2.html [Accessed on 2 November 2003].

Bordwell, D. (1997) 'On The History of Film Style'. Cambridge, Harvard University Press.

Bordwell, D. (1985) 'Narration in the Fiction Film'. London, Routledge

Claxton, G. (1998) *Hair Brain, Tortoise Mind.* London: Fourth Estate.

Cloud Gate Dance Theatre of Taiwan (1999) *Songs of the Wanderers* choreographer Lin Hwai-Min, Taiwan; RM Associates & ARTHaus Musik.

De Landa, M. (1998) 'Virtual Environments and the Emergence of Synthetic Reason'. *In*. Broadhurst Dixon, J. and Cassidy, E. J. (eds.) *Virtual Futures cyberotics, technology and post-human pragmatism.* London: Routledge. pp. 65-76.

Deleuze, G. (1989) *Cinema 2: the time-image* Translator: Tomlinson, H. and Galeta, R. London, The Athlone Press

Deleuze, G. (1986) *Cinema 1: the movement-image.* Translator: B. Tomlinson, H. and Habberjam, London, The Athlone Press.

Deleuze, G. and Guattari, F. (1987) *A Thousand Plateaus: capitalism and schizophrenia.* Translation: Masumi, B., London, Continuum.

Denny, J., Lovell, R. and Mitchell, J. (1996) 'Virtual Space Harmony: The value of Rudolph Laaban's Theory of Space Harmony as a means to Navigate a Virtual Stage Environment (VSE). Institute for Studies in the Arts, [online]. Available from: http://www.siliconatelier.com/lovell/r_papers.html. [Accessed on 11th July 2004]

DV8 Physical Theatre (1994) *Strange Fish,* Choreographer: Lloyd Newson, London: BBC videocassette.

Farley, K. (2003)'Digital Dance Theatre: The marriage of computers, choreography and techno/human reactivity' *Body Space Technology*, [online] 3(1). Available from: http://www.brunel.ac.uk/depts/pfa/bstjournal/index.htm [Accessed on 20th October 2003].

Forkbeard Fantasy (2004) *Shooting Shakespeare*. Devisors and writers Chris and Tim Britton, The Corn Exchange, Brighton UK.

Gaggi S. (1997) *From Text to Hypertext: decentering the subject in fiction, film, the visual arts, and electronic media.* Philadelphia: The University of Pennsylvania Press.

Heim, M (2001) 'The FengShui of Virtual Reality'. *Crossings; ejournal of art and technology* [online]. 1:(1). Available from: http://crossings.tcd.ie/ [Accessed 31/10/2003].

Heim, M (1993) *The Metaphysics of Virtual Reality* Oxford: Oxford University Press.

Hilllis, K. (1999) *Digital Sensations: space identity, and embodiment in virtual reality.* Mineapolis: University of Minnesota Press.

Ihde, D. (2002) *Bodies in Technology*. Minneapolis: University of Minnesota Press.

Landow, G. (1997) *Hypertext 2.0.* London: he John Hopkins University Press.

Lovell, R. (2002)'A Blueprint for Using a Reactive Performance' Space *Digital Performance*, anomalie digital_arts N. 2, Press ATI s.r.l. Rome, Italy. [online] Available from: http://www.siliconatelier.com/lovell/r_papers.html [Accessed on 11th July, 2004]

Lovell, R. (2000) 'Computer Intelligence in the Theatre', *New Theatre Quarterly*, August, [online]. Available from: http://www.siliconatelier.com/lovell/r_papers.html [Accessed on 20th July, 2004]

Lovell, R. (1999) 'Toward Computer Cognition of Gesture', *International Dance and Technology*, [online]. Available from http://www.siliconatelier.com/lovell/r_papers.html [Accessed on 11th July, 2004]

Oxford English Dictionary (1999) Oxford; Oxford University Press.

Random Dance Company (1998) *The Millennarium*, Choreographer Wayne McGregor, London: The Place Videos.

Reaney, M. (1999) 'Virtual Reality and the Theatre: immersion in virtual worlds.' *Digital Creativity*, 10 (3): 183-188.

Rubidge, S. (2002a) 'Digital Technology in Choreography: Issues and Implications'. Published in *the Proceedings of the 17th Annual Symposium of the Dance Society of Korea,* Seoul, Korea, November 6th 2002 Introduction.

Rubidge S. (2002b) 'Identity in Flux: A Practice-based Interrogation of the Ontology of the Open Dance Work'. *In* Preston-Dunlop,V & Sanchez-Colberg, A (eds.) *Dance and the Performative: A Choreological Perspective.* London: Verve Publishing pp. 136 – 163.

Stuart, R. (May 2004) *Compositional approaches for live dance within 'moving' digital scenography: presence, sensation and the time-image*. Seminar paper given at University College Chichester for MA Collaborative Arts course, 12th May 2004.

Stuart, R. (March 2004) *Compositional approaches for live dance within 'moving' digital scenography: ongoing research*. Seminar paper given at University College Chichester for MA Collaborative Arts course, 17th March 2004.

Stuart, R. (2003) *Compositional approaches for live dance within 'moving' digital scenography: an introduction*. Seminar paper given at University College Chichester for MA Collaborative Arts course, 4th December 2003.

Troika Ranch (Date last modified, 2003) Troika Ranch Company website [online]. Available from: http://www.troikaranch.org [Date Accessed: 29th October 2003].

Ultimez Vez (2004) *Blush.* Choreographer, Wim Vanderkeybus. Sadlers Wells, London

Yolande Snaith Theatre Dance (1996) *Swinger* Choreographed by Yolande Snaith, Spring Re-Loaded 2, London; The Place Video Works.

Zbang Dance Company (2000) *'LureLureLure'* Choreographed by Jasmin Vardimon, Spring Re-Loaded 6, London; The Place Video Works.