

VERCOE: When the composer first arrives, he needs to be introduced to the practice of dealing with computer equipment through fairly conventional teletypes, typewriters, and languages. But the object of the studio here has been rather to focus the composer's attention on musical issues by having the computer deal with the more complicated task of interpreting much of his musical idea in the computer language.

Most of the composers who come in here have had some sort of experience in analog equipment and in that event they've been practicing the art of manipulating sounds on magnetic tape. The degree of control they would have over those sounds has normally been limited by the finest with which they can cut tape or hand manipulate analog-controlled equipment, which provided a certain limit on the quality of control that the composer could exert themselves.

What we have here is something very different, where the computer, which is capable of dealing with sound objects down to small milliseconds or microseconds of time, can actually go in and do very, very fine-grained computation on the evolution of timbres, always, of course, under the composer's control. But giving the composer many, many more opportunities to go in and take the responsibility that was formally that of the performer in bringing his conceived music to life.

CHILD: In this movement, I was concerned with chiefly exploiting two capacities of the computer facility. The first of which concerned spatial location and the other of which concerns its almost limitless capacity for complex rhythmic interrelationship among the parts.

VERCOE: The problem that the composer faces when he is using a computer in this way is that of communicating his musical ideas to the machine. Traditionally, composers have communicated with performers via the written page, written music being little more than a skeleton of what the composer really had in mind and the final realization of many of these sounds was actually left to the performer, who would instill the music with all kinds of subtle nuance.

A computer is an instrument, but not being controlled by a performer, it has to be given all the information of subtle nuance explicitly, and therein lies the composer's problem in this case, that he must be much more specific. He's now totally responsible for all of the details of the sound.

What we have managed to do here is to develop modes of communication between the composer and the computing environment that permits information to be transmitted in languages that are much more natural to the composer, in standard music notation, in gestures that are somewhat movements of the other hand, and that allows the composer to get a quality of information in the communication that is very difficult otherwise to express in words.

The languages that become involved here are of two kinds. First of all, languages that will represent musical scores can be, in this case, at any of three levels. One is a strictly numeric language. A second one is one that uses an encoding of music, such as C4 for a quarter note middle C, providing the composer with something that's a little more monic, but is still alphabetic and character-wise in nature. But a third, and the most efficient means of communication, is through a fully fledged score editor of natural music, standard music notation.

The second language that the composer becomes involved in is the representation of the network that actually does the signal processing, the generation of sound involving the specification of sound waves, and finally the multiplication of those sound waves into various shapes that would emerge as individual notes in a fully fledged musical score.

A language that does that has to become involved in painstakingly synthesizing every element of the sound, including all the harmonics that are present in a complex tone, in changing the timbre in the course of the note down to a very strict control by the composer. This then suggests that we provide the composer with languages that enable him to manipulate those sounds in rather musically intuitive terms, in which case we've developed several languages.

Music 11 language is language that affords a composer an intuitive approach to the specification of patching oscillators and filters, but yet in an alphanumeric language. But a cut above that is a language that is, again, graphics based that permits the composer to patch oscillators and filters by moving objects, symbolic objects about on a screen. And that provides a composer with the most instantaneous interaction with the computer.

So the composer learns to, through deft motions on a tablet like this, manipulate objects that are actually on the screen, which represent his signal processing modules.

Probably the most natural way we have of communicating musical ideas to the computing environment is via a keyboard of this type. In this particular instance, when a composer presses a note on this keyboard, the standard music notation representation of that pitch appears on the screen in front of him. And as a result of pressing a sequence of notes, he can enter scores rapidly.

Having set things up on the screen, whether they be specifications of oscillators and filters or specifications of the score, which is to be realized by those oscillation filters in a particular musical context, the composer is probably going to make daily progress on the problem. At the end of each day, he may want some hard copy or some paper copy of where he's got, so that we always preserve the ability or give the composer the ability to copy whatever's on the screen onto the printer here.

I have found that it's very useful to have the visiting composer, who may be technologically somewhat naive, not have to worry too much about the learning of the digital techniques himself, but rather take advantage as he can throughout the academic year of working with an MIT student who works with him in somewhat the guise of a composer's apprentice.

HOFFMAN: I can measure 97. And I hope that you're not tired of my coming back to this so often. The dynamics seem to need some adjustment, possibly a little louder to bring out the pitch definition.

GOLD: Why don't we listen to it? And then we can make some changes.

HOFFMAN: Good idea. It's a little indistinct.

GOLD: I see your point. Okay. Well, why don't we move the cursor here and we can change this F sharp to a forte and change this.

HOFFMAN: Not too much, so that it's [INAUDIBLE]. But I see, yes, that's correct.

GOLD: We'll change this one to a mezzo forte right there.

HOFFMAN: That's perfect. I think that's very good.

GOLD: Okay, shall we hear it again?

HOFFMAN: Yes, I think that is to be preferred. The lower notes are quite distinct and that is, I think, at the peak of the crescendo and the pitch definition is much clearer in this version. I would prefer it.

VERCOE: In watching these composers at work, several things do become apparent as to what things need to be developed next in the future of computer music. One of them is that the communication with computers still needs refinement. Composers, despite the languages we've seen here, do still have difficulty in communicating musical ideas very, very naturally.

A second is that the pursuit of sounds still needs further direction. There needs to be more research on the nature of sounds, on helping the composers pursue a sound or effect that they particularly have in mind.

And in the third category, the composer has in computer music traditionally had to wait quite a long time to get very complex sounds synthesized by a machine that is this small. This being a PDP-11 computer sometimes being shared, time-shared, by six composers at once. It provides the composer with, at times, painstakingly and frustratingly slow return for the time that he is putting it.

Okay, Marcus. I'd like to try this section here from the slow part and transition into the faster section at this tempo. And rework this measure here, so let's try hearing it with the computer, just going from about this measure. I'll cue when there's a downbeat, in addition. Okay.

Okay, coming up now.

Fine. I think that's working much better. That's good. Okay. I don't think you should have any difficulty with that passage now, the way we had the first time. That is a big improvement from the interaction. Good.