



# DSPスーパースクール2004

## ～SuperColliderに耽溺した4日間～

### SuperColliderとは？



SuperCollider 3. テキストが書かれたウィンドウのうち、左が実行した命令のステータスを示すもの、右が記述したプログラム。ここではスクールで配布されたジェームス・マッカートニー氏のサンプル・プログラムを立ち上げており、ステップ・シーケンサーやキーボードを模したウィンドウも表示されている。

音楽／音響分野を中心に、メディア・テクノロジーにおける新しい表現とその手法を探る毎年恒例のイベント、DSPスクール。今年は2月14日、15日、21日、22日の計4日にわたって、『DSPスーパースクール2004』と題し、岐阜県大垣市のIAMASで開催された。初めてSuperColliderをメイン・テーマに据えた、今回のスクールの模様をお伝えしていこう。

例年、CYCLING'74 Max/MSP/Jitterをメインに展開していたDSPスクールだが、今年はSuperCollider(以下、SC)を扱うことになった。SCになじみのない読者も多いと思われるので、どんなソフトかを簡単に説明しておこう。

SCは、Macintosh上で動くリアルタイム音響合成のためのプログラミング言語。CYCLING'74 Max/MSPやNATIVE INSTRUMENTS Reaktorなどと同種のソフトである。SCにはオシレーターやフィルター、エンベロープなどの機能を持つUnit Generatorが300種類以上用意されており、これらを組み合わせてサウンドや楽曲を作成することができる。

ただしSCはテキスト・ベースのプログラミング、つまり文字によるコマンドの入力が必要な点で、グラフィカルにプログラミングが行えるMax/MSP

やReaktorとは大きく異なる。プログラミングの知識があまりなくてもアイデアを形にしやすいグラフィカル・ベースのソフトと比べると、文法や固有のコマンドなど習得しなければいけないことが多く、初学者が取り付きやすいものではない。しかし、柔軟な構造や豊富なUnit Generator群、独自の処理による優れた音質などが評価されており、愛用者も増えつつある。

現在、SCはフリーウェアで、Mac OS X/Linux対応のVer.3が最新バージョン。Webサイト上(sourceforge.net/projects/supercollider)から入手が可能だ。ソース・コードも公開されており、SC開発メーリング・リストで挙げられたアイデアを元に、オリジナル開発者であるジェームス・マッカートニー氏をはじめとする複数のユーザーがアップデートを加えている。

2週にまたがっての週末開催となった『DSPスーパースクール2004』。取材に伺った第2週は、SCのオリジナル開発者であるジェームス・マッカートニー氏(写真①)がメインの講師を務めた。まずは自身のキャリアの紹介からスタートし、SC開発に至るまでの経緯が説明された。SC以前に彼が開発した、レアなソフトウェアが次々登場。資料としてこれらのソフトやSCの初期バージョンも配布されていた。“SuperCollider”という名前は、これらのソフトウェアを融合させたという意味で名付けられたようだ。

続いて、SCのGUI機能を用いた演奏という観点から、さまざまなテクニックが紹介された。SCのプログラミング自体はテキスト・ベースで行うのだが、コントロール用のインターフェースとしてはボタンやスライダーなども用意されている。マッカートニー氏は、上の画面のようなキーボードにサウンドを割り当てて演奏するプログラムやグリッド式のシーケンサーを紹介。これらそのものはよく見かけるタイプのものだが、SCで作成したプログラムを次々とドラッグ&ドロップしてアサインすることができる優れたもの。エレガントかつパワフルなこのプログラムには受講者も関心を集めていた。

さらに、各Unit Generatorを丁寧に紹介。SCのscope機能(生成したサウンドの波形をリアルタイム表示する)を使ってサウンドと波形を示しながら、例えばデイスティーションだけでも、一定以上のレベルをクリップさせる(画面①)、マイナス側の波形を折り返す(画面②)などで10種類以上の手法が示されていた。

その後、話題はC言語でのSC用プラグイン開

発にまで及んだが、全体としては非常に充実した明解なレクチャーだった。

また、第2週には赤松正行氏(写真②)も登場。ネットワーク上でのSCの活用と題し、最近注目を集めているOSC(Open Sound Control)と絡めた解説が行われた。OSCはアプリケーション間でオーディオ/MIDIデータの送受信を可能とするプロトコルで、SC3ではプログラミングを行う部分と実際の音響合成をする機能が分かれており、OSCでコネクティングしているという。また、MaxやReaktorなどのOSC対応ソフトから、SCでのサウンド再生をコントロールすることも可能だ。OSC



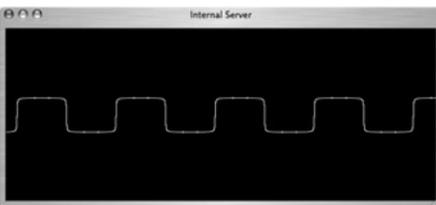
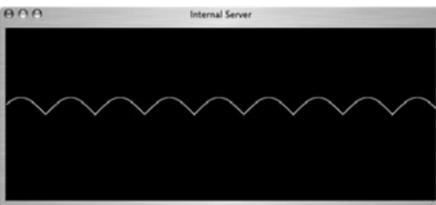
写真① ジェームス・マッカートニー氏の講義

画面① サイン波のマイナス側をプラスに折り返して歪ませた波形。この音を再生しながら画面表示するコードは以下のように記述する。  
{ SinOsc.ar(300, 0, 0.2).abs }.scope(1)

画面② 同じサイン波をクリップによって歪ませた波形で、コードは{ SinOsc.ar(300, 0, MouseX.kr(0.1, 80, 1)).distort \* 0.2 }.scope(1)。このコードでは、マウスの左右の動きで歪み具合を調整できる(画面は最も歪ませた状態)

はLANやインターネットでの利用も可能で、SCで作成した音声合成モジュールを別のコンピューターから無線LANでコントロールするといった試みも行われた。

スクールの最後には、佐近田展康氏の司会進行による『DSPプログラミング大喜利』も行われた(写真③)。落語の大喜利のように、テーマに合わせたサウンドをその場でプログラミングするという、高度なスキルが要求されるものだ。出場者はマッカートニー氏と宇佐見理氏がSCを、三輪眞弘氏と高橋哲男氏がMax/MSPを使用。長嶋洋一氏とYuko Nexus 6氏は、Max/MSPとSC



の両方を使い分けての出場となった。国内屈指のDSPプログラミングの使い手が並ぶ中で、マッカートニー氏も健闘。テーマとして富士山の絵が示された際にも、エキゾチックで端正なサウンドを作り、喝さいを浴びた。

初めてMax以外をメインとしたにもかかわらず、例年並みの受講者数となった今回のスクール。講義資料はWebサイト(dspss.iamas.ac.jp)からダウンロードできる。ファイル内にSCのプログラムと解説が併記されているものも多いので、残念ながら参加できなかった方にも講義内容が理解しやすいと思う。



写真② 赤松正行氏



写真③ DSPプログラミング大喜利。写真は左から三輪眞弘氏、高橋哲男氏のMax組と、当然ながらSCを使用したマッカートニー氏

## Interview : James McCartney

ジェームス・マッカートニー ◆インタビュー



Interpretation : Shigeru Kobayashi

万人のためのツールとは言えませんが  
フレキシビリティを求めるなら  
SCを使ってほしいですね

SCのオリジナル開発者であるジェームス・マッカートニー氏。ソフトウェア・プログラマーとして、また音楽家としてキャリアを重ね、現在はAPPLEにエンジニアとして勤務し、Core Audioの開発に携わっている。今回、初来日を果たした氏に、SCやCore Audioについて話を伺う機会を得た。

■SCを開発したきっかけは？

□自分で作ったサウンドを、柔軟にコントロールできるツールが必要だったからです。SC以前にも音響合成ソフトを作っていたのですが、CPUの処理能力の問題でリアルタイムでの処理はできなかつた。だから合成した音声、外部のサンプラーに読み込ませていたんです。それが1994年に最初のAPPLE Power Macintoshが発表されて、CPUの処理能力が飛躍的に上がった。それがきっかけで、リアルタイムでの音声合成に特化したものとして、SCを作ったんです。

■SCはサウンドが良いという意見も多いのですが、何か特別な工夫があるのですか？

□内部の処理はMSPなどと同じ32ビット浮動小数点なのですが、ある値からある値にパラメーターが変化するとき、滑らかに変化させるための“補間”をうまく行っています。あと、Unit Generatorが300種類くらいありますから、より目的になつたものを使うことで、いいサウンドが得られるのだと思います。

■既成のソフト・シンセも出そろってきている中、SCを使うメリットは？

□SCのポイントは、音響合成能力と同時に、柔軟なコントロールができることにあると思います。一般的なソフト・シンセだと、使いやすいように、コントロールできるパラメーターを限定しているわけです。SCは、もっと複雑なコントロールが必要だという人には最適なツールだと言えるでしょう。

■テキスト・ベースでのプログラミングは、経験とスキルが要求されると思いますが？

□一般的な音楽での楽譜と同様、ある程度の学習が必要でしょうね。柔軟であることを狙っているわけなので、現在のスタイルにしているわけではないことは確かですが、フレキシビリティを求めるなら使ってもらいたいですね。

■Ver.3から、音響合成を定義/コントロールする部分と、実際の音響合成を行う部分とを分けたのはなぜですか？

□処理にはパワフルなコンピューターを使って、コードを書いたりするのはラップトップでもできるようにしたかったんです。複数台でそうしたアーキテクチャーを組めるのが、大きなアドバンテージだと思っています。

■SCがフリーウェアとなったのは、あなたがAPPLEに勤めて、SCの開発に時間が取れなくなったからだと聞いていますが？

□以前はSCだけで生計を立てていたこともありますが、今は本業が別にありますからね。とはいえ、今でもSCの開発には積極的に関わっています。現在は、20人くらいがSCの開発に参加していますが、ものすごく積極的なのは数人です。

■APPLEでの仕事の内容は？

□私の担当は、オーディオ・ファイルの読み込みを扱う部分など、Core AudioのOS側のサービス……Core Audioのコアな部分ですね(笑)。

■Core Audioのように、OS側でオーディオ・ドライバーが用意されていることの利点は？

□そのドライバーに対応したハードやソフトが、OS内で動作することが保証されることがまず第一です。OSが進化していくに従って、アーキテクチャーが変更され、サポートされなくなったり動かなくなったりすることもよくありますが、そういった心配もなくなるわけですね。また、サンプリング・レートの変換やMP3への変換といった機能もCore Audio内部に備えられているので、各ソフトがこれを利用することで、こういった作業が効率に行える。その分、もっと創作的な活動に集中できるわけです。もちろん、レイテンシーが少ないこと、さまざまなソフトやオーディオ・インターフェースが利用できる点なども大きなポイントでしょう。

■今回、日本でこうしたSCをフィーチャーしたイベントが行われて、多くの人が参加したことについては、どうお考えですか？

□あまり驚きはありませんでした。というのも、私のWebサイトへのアクセスも、日本からのものがアメリカ国内に次いで多いんです。まだ使ったことがない人も、こうしたプログラミングで音楽を作ることに興味があるならば、どんどん使ってもらいたいですね。