

# 新世代音楽システム ICOTone の ユーティリティ体系

平田圭二、青柳龍也、小池汎平、阿久津達也、田中英彦  
(東大工学部)

2H-1

## 1. はじめに

我々は新世代音楽システム ICOTone[1]を開発している。ICOTone はワークステーションを基礎とする、ミュージシャン指向のシステムである。

ICOTone の特徴は、ミュージシャンがさまざまな目的(例えば、創作、実験、演奏、楽曲分析など)に使うための**枠組**を提供することである。これは、ICOTone が特定の目的のみを指向した閉じたシステムではなく、拡張性やcustomizability を考慮した開かれたシステムであることを意味する。このため、ICOTone では、音楽情報を分類し、それらの情報を処理するためのユーティリティを、ユーティリティ群として系統的に用意する予定である。ミュージシャンは、それらのユーティリティを自分の目的に応じて自由に組み合わせて使うことができる。

本稿では、まず、ICOTone で扱う音楽情報を分類し、次にユーティリティ群の体系を紹介する。

## 2. ICOTone で扱う音楽情報

### 2.1 音楽情報

物理的な空気の振動である音は、人間が能動的にその音から何かを感じとったときに、はじめて音楽となる。ある人は音から音程を認識するかもしれないし、ある人はリズムを感じとり、ある人は気持ちを高ぶらせ、ある人は作曲者の名前や曲名を想起するかもしれない。このように、音に基づいて人が認識するさまざまなもの、もしくは、それと等価なものを音楽情報と呼ぶ。

人間が音から感じとる音楽情報は個人差が著しく、例えば絶対音感のある人とない人では音程の感じ方は異なるだろうし、知っている曲の量が違えば曲名を想起する回数も違うであろう。しかし現時点では、ICOTone は調性音楽を対象として開発を進めており、ICOTone システムを使うミュージシャンに話を限れば、音程やリズムの感じ方には共通の認識があると考えて良いだろう。

### 2.2 表現形式

同一の音楽情報を表現する際、その表現形式は一意に定まらない。例えば、「八分音符」という情報は、計算機の中であるビット列で表現することもできるし、印刷された楽譜の上で、♪という形で表現されるかもしれな

い。

同一の音楽情報に対しいくつも存在する表現形式を大きく二つに分けることにする。計算機で扱うことのできるmachine readableな形式と、人間が読むことのできる形式の二つである。

ICOTone システムは、ワークステーション上に構築される。このため、ICOTone で扱う音楽情報はmachine readableでなければならない。

### 2.3 音楽情報の記述量

人間が音楽を認識するレベルは様々であり、その認識のレベルにより音楽情報の記述量は大きく異なる。ICOTone では、この記述量の多少により音楽情報の階層化を行う。人間は音楽を聞いている時、多くの記述量を要する情報から必要な情報を抽出する作業を行っている。また音楽を創造している時は、記述量の少ない情報に自分なりの情報を付け加える作業を行っている。

### 2.4 ICOTone で扱う情報

ICOTone では記述量によって階層化されたmachine readableな音楽情報を扱う。

図1に、ICOTone で扱う音楽情報を示す。四角で囲ま

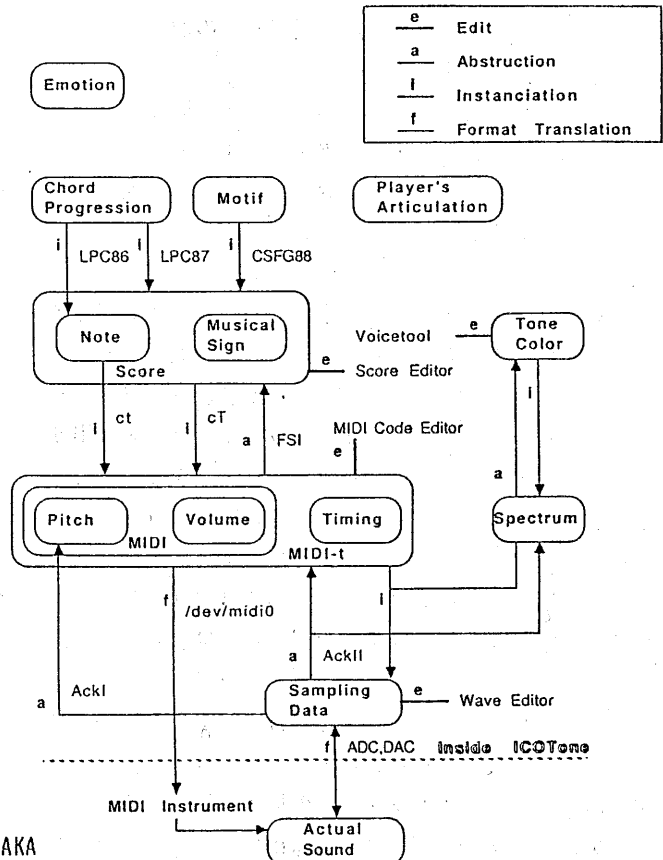


図1 ICOTone で扱う音楽情報

れたものが音楽情報である。図の上の方は記述量が少く、下の方は多い。また矢印はユーティリティを意味する。実際の音に最も近いのがサンプリングデータであり、記述量は最も多い。その上には、音程、音量、周波数スペクトルそれぞれの時間的変化や、タイミングの情報が位置する。これらの内、音程、音量に関してはMIDIコードという形式で規格化されている。我々はそれにタイミング情報を加えたMIDI-tコードを提案した[2]。印刷された楽譜に近いレベルの情報が楽譜情報である。楽譜情報は、音符に関する情報と奏法指示の情報に分けてとらえている。さらにこの上には、コード進行、モチーフ、演奏家の奏法情報等がある。最も上のレベルには人間の感情というレベルが存在する。

### 3. ICOTone で提供するユーティリティ群

#### 3.1 ユーティリティの分類

ICOTone のユーティリティは、表現形式と記述量の二つのパラメータで分類できる。ICOTone の殆どのユーティリティは、ある音楽情報を入力し、処理した後、新しい音楽情報を出力する。その時の入出力の表現形式が同じかどうかと、入出力の音楽情報の記述量がどう変化するかでユーティリティが分類できる(表1)。

表1. ICOTone におけるユーティリティの分類

ユーティリティ のタイプ	入出力データ の表現形式	音楽情報 の記述量
編集	同じ	同じ
抽象化	異なる	大→小
具体化	異なる	小→大
形式変換	異なる	同じ

編集ユーティリティは、音楽情報の表現形式や記述量を変えずに、内容を変更する。ICOTone の編集ユーティリティのマンマシンインタフェースでは、ワークステーションの技術を応用し、ウインドウ、マウス等を使用する。これに属するユーティリティとして、現在までに楽譜エディタ[5]、音色エディタ[6]が開発されている。

抽象化ユーティリティは、記述量の多い音楽情報から、望みの情報を取捨選択する。表現形式は異なったものとなる。抽象化ユーティリティの例としては、AckII[9]とFSI[3,7]がある。AckIIは、サンプリングされたデータから音程、音量、タイミングの情報を抽出する。FSIはMIDI-t形式で表現された、音程、音量、タイミングの情報から、音符の情報と奏法指示の情報を抽出する。

具体化ユーティリティは、記述量の少ない音楽情報に情報を付加し、より実際の音に近い音楽情報を生成する。

この情報の付加は、通常人間の行っている創作活動に相当する。我々は具体化ユーティリティにより、創作活動には必要なことであるが、本質的でない決まり切った仕事の部分を支援する。このユーティリティの1つであるLPC86,87[11,8]は、コード進行を実際の音符に変換する作業のうち、規則として扱える部分を支援する。またct[4]は、音符の記述からMIDI-tへの変換作業を支援する。

形式変換ユーティリティは、内容を保存したまま情報の表現形式だけを変換する。主に、入出力のユーティリティがこれに相当し、例えば/dev/midi0[2]は、計算機内のMIDI-t形式の情報を、実際のMIDI楽器が受信できる形式に変換する。またADC,DAC[10]は、実際の音とサンプリングデータとの間の変換を行う。

#### 4. 終わりに

ユーティリティを設計するためには、扱う音楽情報の表現形式を決定しなければならない。現在、一部の音楽情報の表現形式のみ決まっている。今後は各音楽情報の表現形式の詳細を決め、ユーティリティの設計を進める予定である。

また、現在までに開発されているユーティリティはバッチ処理的なものが多い。しかし音楽には時間芸術という側面が強いため、リアルタイム性を十分考慮したユーティリティを開発する必要がある。

#### < 参考文献 >

- [1] 平田他, "新世代音楽システムICOToneの全貌", 第33回情報処理学会全国大会 5N-5
- [2] 青柳, 小池, "ICOTone-/dev/midi0", 第33回情報処理学会全国大会 5N-6.
- [3] 青柳, "ICOTone-採譜プログラムFSI", 第33回情報処理学会全国大会 5N-9.
- [4] 平田, "ICOTone-自動演奏プログラム concurrent 楽", 第33回情報処理学会全国大会 5N-7
- [5] 金井, 斉藤, "ICOTone-楽譜エディタ", 第33回情報処理学会全国大会 5N-8.
- [6] 小池, "ICOTone-音色エディタ", 第33回情報処理学会全国大会 5N-10.
- [7] 青柳, 田中, "ICOTone~FSI on PSI", 第34回情報処理学会全国大会 2H-2.
- [8] 平田他, "ICOTone~ジャズ和音生成プログラム LP C87", 第34回情報処理学会全国大会 2H-4.
- [9] 阿久津他, "ICOTone~メロディ抽出プログラム AckI", 第34回情報処理学会全国大会 2H-5.
- [10] 鈴木他, "ICOTone~音楽ワークステーション用大容量Wave Memory", 第34回情報処理学会全国大会 2H-6.
- [11] 平田他, "新世代音楽システムICOTone -立派な和音LPC86", 日本ソフトウェア科学会第3回全国大会.