

MOMI: 音楽メタ情報記述のためのフレームワーク

平田 圭二^{†1} 松田 周^{†2} 青木 忍^{†3} 浜中 雅俊^{†4} 梶 克彦^{†5} 長尾 確^{†5}

^{†1} NTT コミュニケーション科学基礎研究所 ^{†2} デジタル・アート・クリエーション ^{†3} COXS
^{†4} 科学技術振興機構 さきがけ研究員 ^{†5} 名古屋大学

hirata@bri.ntt.co.jp

あらまし

本稿では、楽曲データに対して音楽理論 GTTM の分析結果をアノテーションとして記述する方式と現在進めている音楽メタ情報記述のフレームワーク (MOMI) の設計について述べる。MOMI の目的は、付与されたアノテーションの再利用性を高め、音楽情報の作成・蓄積・交換・流通を促進することである。そのため解決すべき主な課題として、既存技術との連続性を考えてデファクトスタンダード規格を採用すること、コンテンツの構造と意味に関するアノテーションを区別すること、コンテンツ制作のプロセス (操作の適用) に関する情報が再利用できることなどを挙げた。これらに対し、GTTM 分析結果を記述するアノテーションと制作プロセス再利用のためのアノテーションを提案する。今後近い内に、MOMI の XML Schema とそれに準拠した楽曲データ及びアノテーションの配布を考えている。

MOMI: Music Meta Information Description Framework

Keiji Hirata^{†1} Shu Matsuda^{†2} Shinobu Aoki^{†3}
Masatoshi Hamanaka^{†4} Katsuhiko Kaji^{†5} Katashi Nagao^{†5}

^{†1} NTT Communication Science Laboratories ^{†2} Digital Art Creation ^{†3} COXS
^{†4} PRESTO, Japan Science and Technology Agency ^{†5} Nagoya University

1 はじめに

我々はこれまで音楽 (旋律, 和声, リズム) の形式化 [HA 03], 形式化の基礎である音楽理論 GTTM に準拠した自動分析器の実装 [HHT 05], 応用システムの構築 [KHN 05] に取り組んできた。これら一連の研究開発を通じて、楽曲データおよびそのメタデータの標準的な記述形式の必要性を痛感した。もしそのような記述形式があれば、研究者や利用者のコミュニティ全体に次のような利点がもたらされるであろう: 音楽の構造や意味を考慮した処理が容易になる、データやシステムの相互運用性が高まる (複数の応用で共通化できる)、楽曲データの作成・蓄積・交換・流通が促進される。

このような問題意識に沿って、これまで楽曲データおよびそのメタデータの標準的な記述形式の提案がいくつか行われている (表 1) [Nagao 03]。

Table 1: 従来のコンテンツとアノテーション

	役割	記述形式
アノテーション	構造, 意味, オントロジ	XML, XLink, RDF, RDFS
コンテンツ	記号, 信号	XML

コンテンツは人が直接的に制作、受容するものであり、それそのものだけで利用可能という特徴がある。例えば楽曲はコンテンツである。コンテンツとしての楽曲は音の時系列を記述すれば十分なので、大まかに、時刻とイベントの組の集合で記述する方式と、時刻順にイベントを整理して記述する方式が考えられる。いずれも木構造つまり XML で記述できる。

一方、アノテーションはコンテンツに対するメタ情報であり、一般に人手が介入しないと得るのが難しいものを指す。アノテーションの例には、その楽曲の曲名、作曲者、作曲年などの書誌的な情報、小節や楽節など楽曲の構造、他のコンテンツとの関係、ユーザの嗜好やコメントなどがあり、再利用を前提に付与される。

楽曲データの記述において、コンテンツやアノテーションを効率よく正確に記述できる記述形式として MusicXML, WEDELMUSIC, MPEG7 などが提案され、すでにある程度利用されている。しかし、音楽理論に基づくアノテーションの記述や応用システム構築の経験と、付与されたアノテーションが十分に再利用されているかどうかの観点から、これら既存の記述形式にはまだ改善の余地があると考えられる。例えば、アノテーションとして記述できる内容が書誌的な情報に限定されている、ユーザが意図するコンテン

ツの一部分を参照してアノテーションを付与するのが難しい、またその付与されているアノテーションを検索するのが難しい、楽曲データを制作するプロセス(操作、処理)に関するアノテーションが付与できず、制作プロセスの再利用ができない、などである。

これらの問題点を解決するために、関係が表現できるような記述形式を用いてコンテンツを構造化するアノテーションと制作プロセスを表現するアノテーションを導入する。

本稿の構成は以下の通り。第2章ではアノテーションの付与、利用の観点から既存の音楽情報記述形式の問題点を整理し、構造化アノテーションと意味的アノテーションの区別がアノテーション再利用にとって効果であることを述べる。第3章では、音楽理論 GTTM に基づく構造化アノテーションを採用入れた音楽メタ情報記述のフレームワーク MOMI を提案する。さらに、制作プロセス再利用のため構造化アノテーションも提案する。第4章で、現在の開発状況や今後の課題等を述べる。

2 音楽メタ情報記述のためのフレームワーク

2.1 要請

我々は、音楽メタ情報記述のフレームワークに対し以下の点を要請する。また MOMI の設計において生じるトレードオフも整理する。

既存の音楽記述形式との連続性を重視する立場からの要請 デファクトスタンダードの利点を享受し、既存の規格と連続性のある記述方式を目指す。まず我々はコンテンツ(楽譜)記述方式として MusicXML を採用する。その理由は、すでに MusicXML の普及が進み MusicXML 利用者のコミュニティが確立されていること、MusicXML に関連するツール群が充実していること、それゆえ改良のスピードも早いからである

コンテンツにアノテーションを付与する立場からの要請 アノテーションの付与は、ユーザがアノテーションを付与したい部分を同定(参照)するステップと、コメント本体と参照を関連付けるステップから成る。アノテーションを付与したい部分の同定とは、コンテンツ構造に関するユーザの認識に基づいて部分と全体、部分と部分を区別することである。例えば小節、楽節、類似した旋律の繰り返しの先頭、サビの区間などがある。しかし、既存の音楽記述形式を用いると、小節や楽節は比較的容易に同定できるが、類似した旋律の繰り返しの先頭やサビの区間などは一般に適切に参照することが難しい。

音楽理論 GTTM の分析結果であるグルーピング構造、拍節構造、タイムスパン木もユーザの意図するコンテンツ構造として有用であるが、既存の音楽

記述形式を用いて参照するのは難しい。そして、これらの情報は書誌的情報と異なり、個々の音どうしの関係(ネットワーク)、あるいは音の集合どうしの関係として表現される場合が多い。

あるユーザがアノテーションを付与しようとした時、他のユーザあるいは以前のユーザ自身がそのコンテンツ部分にどのようなアノテーションを付与したのかを知りたい場合がある。この時、コンテンツ(MusicXML ファイル)中の個々のオブジェクトに関する識別子から、そのオブジェクトへのアノテーション(一般に複数種類かつ複数通り)を検索する機能が必要となる。

複数人が独立に同時にコンテンツにアノテーションを付与でき、またすでに付与されたアノテーションを再利用するためには、一旦コンテンツに付与して公開したアノテーションには変更を加えられないよう制限する必要がある。MDS などの認証方式を利用することが考えられる。

コンテンツやアノテーションを利用する立場からの要請 アノテーションの目標の1つが人手を介して生成された貴重な情報の再利用であるとするなら、生成結果としての楽曲データを再利用するだけでなく、楽曲データを制作するプロセス(操作、処理)の再利用も考慮に値するであろう。そのために、制作プロセスに関するアノテーションが付与できるような記述形式が望まれる。

ユーザがあるコンテンツを受容する時も、付与する立場と同様に、コンテンツの部分に付与されているアノテーションを知りたい場合がある。この時も、コンテンツ中の個々のオブジェクトに付与されたアノテーションを検索する機能が必要となる。

2.2 アノテーション再考

アノテーションとして付与されるコメントは、本来そのコンテンツ全体に対するものであったり、その一部分に対するものであったりする。しかし前述したように、コンテンツ構造に関する認識に基づいて部分と全体、部分と部分を区別する適切で一般的な方法が存在しないため、従来のアノテーションの枠組ではアノテーションの役割に構造、意味、オントロジの3つが混在することとなった(表1)。

我々は、構造化と意味の2層のアノテーションを区別する(表2)。役割の異なるアノテーションは区別して管理する方がより効率良く適切に機能させることが可能になると考えるからである。

Table 2: MOMI におけるコンテンツとアノテーション

	役割	記述形式
意味的アノテーション	意味、 オントロジ	RDF, RDFS
構造化アノテーション	構造	RDF, XLink
コンテンツ	記号, 信号	XML

構造化アノテーションはコンテンツの構造に関するメタ情報を表す。MOMIで導入する GTTM 分析結果は構造化アノテーションの1つであり、文脈依存性が低つまり個人ごとの差異は少ない。他の構造化アノテーションとしては例えば、楽曲の曲名、作曲者、作曲年などがある。構造化アノテーションは、コンテンツ全体あるいはコンテンツを構成する(複数の)部分を引数とするような述語として表現され、述語は引数間の関係なので RDF や XLink で記述できる。

意味的アノテーションはコンテンツと構造化アノテーションに対するメタ情報であり、同種で複数のアノテーションに言及して比較対照したり、異種で複数のアノテーションに言及して関連付けることで得られる構造や意味等を指す。意味的アノテーションはオントロジの一種と見なすこともできる。例としては、ボキャブラリの宣言(スキーマ)、個人ごとに異なる嗜好(比較対照)、変化からグルーピング境界を判定するルールの下位ルールに音程やリズムの変化を調べるルールがある(クラス階層)、タイムスパン木の primary 属性の逆向き属性として stable 属性がある(プロパティ関係)などがある。意味的アノテーションは、コンテンツ全体、コンテンツの一部分、構造化アノテーションに関する述語を引数とするようなメタ述語として表現される。その記述には RDF や RDFS を用いることができる。

一般に、構造化アノテーション付与より意味的アノテーション付与の方が人手が介入する程度は高い。

我々は MOMI における構造化アノテーションをむしろコンテンツの一部と見なした方が合理的ではないかと考えている。それは、コンテンツ構造を共通化することが意味的なアノテーションの再利用性を高めるからである。

3 MOMI の構造

前節で述べた要請の内、GTTM 分析結果の表現と制作プロセスの再利用に対する解決法について述べる。本節で導入する GTTM 分析結果を記述するアノテーションと制作プロセス再利用のためのアノテーションはともに構造化アノテーションである。

3.1 XML ファイル中要素の参照

我々はコンテンツ記述形式として MusicXML を採用した。その MusicXML ファイル中では基本的に、楽譜上に表示されるオブジェクト(音符、休符、小節など)が時間順に出現する。それらオブジェクトに任意にアノテーションを付与するためには、個々のオブジェクトを参照できなければならない。一般に XML ファイル中の要素を参照するためには XPointer が用いられる。MusicXML ファイルの場合、下のような XPointer を用いればよい。

```
<moml:note xlink:href="#xpointer(
  //part[@id='P1']/measure[@number='4']
```

```
/note[3])"/>
```

この例では、パート 1、第 4 小節中 3 番目に出現する音符を参照している。

ある要素に到達する XPointer の記述は複数通り存在するので、ここで問題となるのは参照の識別子としての一意性である。識別子の一意性を保証するために、XPointer 仕様のサブセットのみを利用するか、MusicXML ファイルを XML の木構造として辿るような記述方式を採用する必要がある。

この識別子の一意性を実現するのがアノテーションサーバである。アノテーションサーバは、問合せとして個々のオブジェクトを参照する URI を受け付け、そのオブジェクトを参照するアノテーションを返す。この時、URI は一般に XPointer によって記述されているが、アノテーションサーバ内でその XPointer 記述が正規化される。

3.2 GTTM 分析結果の表現

MOMI では、GTTM の分析結果をほぼそのまま表現する形式を採用した。

グループ構造: グループ構造の表現において、1つのグループは下のように group タグで囲まれた要素として表現され、その要素中に MusicXML ファイルの個々の音符への参照が含まれる。

```
<group>
  <moml:note xlink:href=
    "#xpointer(/.../note[2])"/>
  <moml:note xlink:href=
    "#xpointer(/.../note[3])"/>
</group>
```

グループ境界で成立する規則は下のように applied タグで表現され、成立した分だけ規則が列挙される

```
<group>
  ...
</group>
<applied rule="2a"/>
<applied rule="6"/>
<group>
  ...
</group>
```

group 要素の中に group 要素が再帰的に出現できる。グループ構造を記述する XML 仕様を GroupingStructureXML と呼ぶ。

拍節構造: 拍節構造は、各 tactus 毎に metric 要素を割り当て、もしある時刻において規則が成立していた場合は group 要素と同様に下のように applied 要素で表現する。その tactus 上に音符が乗っている場合、metric 要素内にその音符への参照(moml:note 要素)を置く。

```

<metric dot="4" at="0.0">
  <applied level="0.125" rule="3"/>
  <applied level="0.125" rule="5c"/>
  ...
  <applied level="1.0" rule="5e"/>
  <momi:note xlink:href="..note[1].."/>
</metric>
<metric dot="1" at="0.125"/>
<metric dot="2" at="0.25">
  <applied level="0.125" rule="3"/>
  <applied level="0.25" rule="3"/>
  <momi:note xlink:href="..note[2].."/>
</metric>

```

ここで、dot 属性はその tactus を強拍とする拍節長が何段あるかを表し、at 属性は、四分音符の長さを 1.0 として、その tactus の楽曲先頭からの生起時刻を表す。level 属性と rule 属性は、どの拍節長レベルでどの規則が成立しているかを表す。level の値は、四分音符が 1.0 であり、16 分音符が 0.25 (= 1.0/4) である。拍節構造を記述する XML 仕様を MetricalStructureXML と呼ぶ。

タイムスパン木: タイムスパン木の 1 つのノードは下のような ts 要素で表現される。

```

<ts>
  <head>
    <momi:note
      xlink:href="..note[1].."
      xlink:type="simple"/>
  </head>
  <primary>
    ...
  </primary>
  <secondary>
    ...
  </secondary>
</ts>

```

この head, primary, secondary の 3 つの要素はそれぞれ、タイムスパン木のヘッド、主枝、副枝の要素を表す。主枝、副枝の中には再帰的に ts 要素が出現する。葉は head 要素のみを含む ts 要素として表現される。

head 要素中にはヘッドとなる音への参照が含まれる。属性 xlink:type は参照の型を表し、上の例の simple は最も単純な一方方向の参照であることを表している。GTTM ではヘッドの種類として ordinary, fusion, transformational があり、head 要素の type 属性として記述されるが、上の例ではデフォルト値 ordinary なので type 属性は省略されている。タイムスパン木を記述する XML 仕様を TimeSpanTreeXML と呼ぶ。

時間構造: MOMI には含まれるが GTTM には含まれない音楽構造として時間構造 (temporal structure) がある。これは、イベント生起時刻列の抽象化・具体化

に関する構造である [HA 03]。ある音は先行音 (pred) と後続音 (succ) の間で生起すると解釈する時、pred か succ のいずれかはタイムスパン木の直上の primary 枝の音と等しく、そうでない方は当該音の周囲の適切な音から選ばれる。前者を stable と呼び、後者を unstable と呼ぶ。

```

<TemporalStructure xmlns=...
  <unstable xlink:type="extended">
    <momi:note
      xlink:href="..note[1]"
      xlink:type="locator"
      xlink:label="original"/>
    <momi:note
      xlink:href="..note[3]"
      xlink:type="locator"
      xlink:label="unstable"/>
    <direction
      xlink:from="original"
      xlink:to="unstable"
      xlink:type="arc"/>
  </unstable>
  <unstable xlink:type="extended">
    ...
  </unstable>
  ...
</TemporalStructure>

```

ある音にとっての stable/unstable な音が他の音にとっても stable/unstable な音になる場合があるので、一般に時間構造はネットワーク構造である。よって、MOMI では時間構造を表現するのに XLink の拡張リンク (extended) を用いる。上の例では、当該音 note [1] にとって note [3] が unstable な音であることを表すために、original である note [1] から note [3] の向きにリンク (arc) を張っている。現在時点では、RDF でないと記述できないような高階で複雑なリンク構造がないため XLink を用いている。時間構造を記述する XML 仕様を TemporalStructureXML と呼ぶ。

3.3 制作プロセスの再利用

MOMI では、楽曲の変換や処理自体も再利用可能とするため、楽曲変換や処理を表現するアノテーションとコンテンツを導入する (図 1)。図中、四角はコンテンツを表し、三角はアノテーションを表す。引数、操作 (操作の定義やパラメータ)、結果のコンテンツは各々 URI で参照されており (arg. op., res.), depend アノテーションがこれら 3 つのコンテンツを関連付ける。ここで、操作コンテンツが操作の実体を表現し、depend アノテーションがその操作の適用順序を表現する。これは操作のようなメタレベル情報を“コンテンツ化”することに相当する。操作とその適用が明示的にデータ構造として表現されることで制作プロセスの再利用が可能になる。

操作によっては複数の引数や複数の結果を参照する場合がある。例えば、ある検索を行って、適切さで順序付けられた複数の結果を得たような時、適切さ

4 おわりに

音楽メタ情報記述に対する要請とその解決法は、現在もまだ検討中かつプロトタイプ実装中である。

MOMI では、MusicXML で記述された楽曲に対する GTTM 分析結果をアノテーションとして付与したが、その時間構造は単純なネットワーク構造なので現在は XLink を採用している。しかし、今後は多様で複雑なアノテーションを表現できるようになることに備え、また関連する高機能ツール群の充実度を考慮し、RDF に切り換える予定である。

応用システムの観点から必要な機能を幅広く検討し MOMI に統合していくことを考えている。特に MusicXML 形式以外のコンテンツ、例えば MP3, WAV, QuickTime などのストリームデータ形式との連携については別稿にて議論する予定である。

本発表より近い内に、MOMI の XML Schema とそれに準拠した楽曲データ及びアノテーション約 100 曲の配布準備中である。

謝辞： MOMI に準拠した楽曲データの制作に関し JAIST 東条敏教授からのサポートを得た。

参考文献

- [Good 00] Michael Good, Representing Music Using XML, In *Proceedings of ISMIR 2000*.
- [HHT 05] 浜中, 平田, 東条, ATTA: exGTTM に基づく自動タイムスパン木獲得システム, IPSJ 研究報告, 2005-MUS-61.
- [HA 03] Hirata, K. and Aoyagi, T.: Computational Music Representation based on the Generative Theory of Tonal Music and the Deductive Object-Oriented Database, *Computer Music Journal* Vol.27 (3), pp.73-89, The MIT Press (2003).
- [KHN 05] 梶, 平田, 長尾, コミュニケーションメディアとしてのプレイリストを目指して, In *Proceedings of FIT 2005*.
- [Nagao 03] Katashi Nagao, Digital Content Annotation and Transcoding, Artech House, 2003.

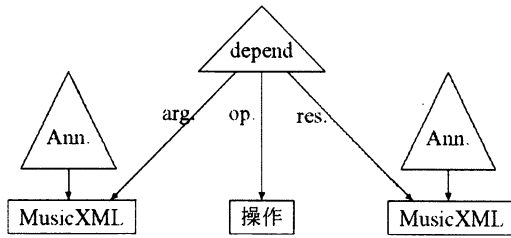


Figure 1: depend アノテーションと操作コンテンツ

の順序を表現するために、その複数の結果間に適切さのリンクを張ることがあろう。この場合、depend アノテーションはネットワーク構造となる。

depend が直接参照するのはコンテンツであるが、MOMI ではコンテンツ (MusicXML) からそれを参照しているアノテーション (Ann.) を検索できるので、depend がアノテーションを直接参照する必要はない。depend アノテーションは、制作プロセスの構造についての情報で文脈依存性が低いつまり個人ごとの差異が少ないので、構造化アノテーションと見なせる。

2つの楽曲(旋律)が与えられて、一方からもう一方へ段階的に変化する楽曲を生成するモルフィング操作の例を図2に示す。モルフィング操作の引数と

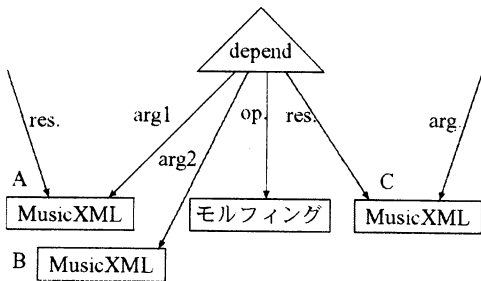


Figure 2: モルフィング操作の例

なる楽曲 A, B は arg1, arg2 というリンクで、操作の実体は op. で、結果の楽曲 C は res. で参照されている。depend アノテーション中には、実行時刻、実行者や補助的なパラメータ等の情報が保持される。引数の楽曲 A は、先行する操作の depend から res. のリンクで参照されており、結果の楽曲 C は後続の操作を表現する depend から arg. リンクで引数として参照されている。これら arg. リンクや res. リンクが操作の適用順序を表現している。