

オーサリングシステム Extempore — エージェント記述

3S-3

大島 昭* 小島 透* 井上 雅裕* 五味 弘* 長坂 篤**

*(株) 沖テクノシステムズ ラボラトリ

**沖電気工業 (株)

1 はじめに

我々が開発しているオーサリングシステム Extempore^[1] のエージェント記述について述べる。

Extempore では、マルチメディア・アプリケーションを作成するときに動画、音声、描画、テキストなどの各々のメディアオブジェクトを静的なオブジェクトとして扱うのではなく、相互に協調して自律的に動作するエージェントとして扱う。このエージェント指向機能を持ったモデルを提供している。

このモデルによって、メディアオブジェクトの全体の関係をすべて指示するのではなく、部分関係やエージェントの集まりである環境のみを記述することにより、各々のメディアオブジェクトの動作を間接的に指示して、マルチメディア・アプリケーションを記述できる。

本稿では、Extempore のエージェント記述をオブジェクト指向と比較し、関係記述を中心に述べる。

2 エージェント

Extempore ではメディアオブジェクト間の全体の関係を記述するものとして、タイムチャートを用意している^[2]。しかし、すべての全体の関係を記述するのでは複雑になり、大規模なアプリケーションの作成は困難になる。

そこで各々のメディアオブジェクトが、環境や他のメディアオブジェクトの状態などから自分で判断して、動作を自律的に行なえるようになれば、大規模なアプリケーションにおいても作成が容易になる。

そのために各々のメディアオブジェクトを自律的に動作するエージェントとして扱うモデル^[3]を提供し、エージェント間の部分関係に注目してエージェントの動作を記述する GUI を与えた。

2.1 オブジェクトとエージェント

メディアオブジェクトを静的なオブジェクトとして捉え、そのオブジェクトに関係するすべての動作を記述し、全体の関係を記述しなければならない。一方、エージェントとして捉え、エージェントの集まりである環境の記述及びエージェントの動作を記述するだけで全体の関係やエージェントの実行を記述することなく、アプリケーションが記述できる。

以下に例を挙げる。

例 大きい魚と小さい魚

水槽の中に大きな魚と小さな魚が数匹いる。大きな魚は小さい魚が近くにいれば、追いかける。小さい魚は大きな魚が近くにいれば、逃げる。その他のときはランダムに泳ぐ。

オブジェクト記述

大きい魚の動作記述	すべての小さい魚の位置と自分の位置を調べて、近ければ、追いかける。
小さい魚の動作記述	すべての大きい魚の位置と自分の位置を調べて、近ければ、逃げる。
全体関係の記述	水槽にいる魚の動作を順番に実行する。これを繰り返す。

エージェント記述

大きい魚のインタプリタ	小さい魚が近くにいるという関係が成立すれば、追いかける。
小さい魚のインタプリタ	大きい魚が近くにいるという関係が成立すれば、逃げる。
関係記述	大きい魚と小さい魚が近いという関係があるかどうか

オブジェクト記述では、自分以外のすべての魚との関係を各々が調べて、逃げるなどの動作を行なうようにする。これをすべての魚が行なう。

一方、エージェント記述では、各々の魚は自分に対して近いという関係があれば、逃げるなどの動作を行なう。

水槽において、近いという関係があるかをパターンマッチして調べ、そのようなパターンがあればイベントを発生する。一方、エージェントは自分に関係したイベント（関係）があれば、自分が持つインタプリタに従って動作する。

2.2 エージェント記述

Extempore ではメディアオブジェクトの属性をオ

Authoring System "Extempore" — Agent Description
Akira Ohshima*, Toru KOJIMA*,
Masahiro INOUE*, Hiroshi GOMI*
and Atsushi NAGASAKA**

*Oki Technosystems Laboratory, Inc.

**Oki Electric Industry Co., Ltd.

プロジェクト記述の GUI「プロフィール」[2]で記述する。プロフィールには、メディアオブジェクトのサイズ、位置などの属性やマウスクリックやキー入力に対して、どのように反応するかなどの情報を記述する。

また、そのオブジェクトがアクティブになったときの動作、イナクティブになったときの動作、アクティブ中の動作も記述する。

メディアオブジェクトをエージェントとして容易に記述するためにエージェント間の関係に注目した。特定または不特定のエージェント、環境間の関係を記述することにより、全体の関係を書かずに、マルチメディア・アプリケーションが記述できる。

以下の章で関係記述について報告する。

3 関係記述

3.1 関係

関係には、エージェント間の関係、環境との関係があり、関係の対象となるエージェントには特定のエージェント、不特定のエージェントがある。

不特定のエージェントを具体化するために、パターンマッチを行なう。具体化されたエージェント間で関係が成立する。

以下に関係の例を挙げる。

例 ; (1) 特定エージェントと不特定エージェントの関係
 ; (近い 大きい魚 A 任意の魚 fish [水槽])
 ; (2) 環境との関係
 ; (水がきれい 大きい魚 A 水槽)

(1) の関係は、特定のエージェント「大きい魚 A」、不特定のエージェント「任意の魚 fish」が関係「近い」にあるかどうかの関係である。環境は「水槽」であり、記述されていないときは大きい魚 A の属する環境が既定となる。

(2) の関係は、特定のエージェント「大きい魚 A」にとって環境「水槽」の水がきれいかどうかの関係である。

3.2 関係/動作記述

関係記述では、これらの関係と関係が成立した時の動作を記述する。動作にはリンク動作とスクリプト動作が記述できる。また、関係と動作にはそれぞれプライオリティが設定できる。

複数の関係が成立するときは、制御の簡単化のために、プライオリティ優先先着順で動作するようにしている。但し、スクリプト言語により細かなイベントキュー制御も行なえるようにしている。

関係記述の GUI を図 1 に示す。

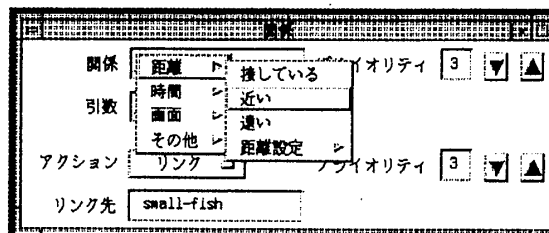
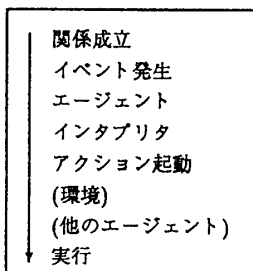


図 1 関係記述の GUI

4 エージェント動作



関係記述による動作は、不特定のエージェントに対する関係をパターンマッチして関係が成立または特定のエージェントの関係が成立したときに、対象になるエ

ージェントのインタプリタが起動され、そのアクションの処理が行なわれる。この様子を以下に示す。

5 おわりに

本稿ではオーサリングシステム *Extempore* のエージェント記述について報告し、「関係」に注目にしたエージェント記述を提案し、関係に注目して記述することにより容易に自律的な動作が記述できるようになった。

他の動作記述では、タイムラインによる動作記述はすべてのメディアオブジェクトの動作を時間軸に表現しなければならず、フローチャートによる動作記述はすべてのオブジェクトの状態変化を記述する必要があり、カード&スタックは複数のオブジェクトの同期をとるのが困難である。

今後の課題としては、一般的なすべての動作をこのエージェント記述で記述できるように関係エージェントを拡張することがある。

参考文献

- [1] 井田、新谷、長坂、五味: “オーサリングシステム *Extempore* — 概要”, 情報処理学会第 43 回全国大会, 3S-1
- [2] 新谷、長坂、五味: “オーサリングシステム *Extempore* — メタファ”, 情報処理学会第 43 回全国大会, 3S-2
- [3] 五味、水崎、山田、長坂: “オーサリングシステム *Extempore* — スクリプト言語”, 情報処理学会第 43 回全国大会, 3S-4
- [4] 丸一、市川、所: “自律的エージェントからなる組織の計算モデルと分散協調問題解決への応用”, 情報処理学会論文誌 Vol.31, No.12, 1990