インタラクティブアニメーション記述言語とその処理系「AV-Script」

1D−8

濱田浩行　阪本清美　John Zurowski
松下電器産業（株）メディア研究所

1　はじめに

コンピュータとＡＶ機器の融合が進む中で、アニメーションはその両方の性質を合わせ持つメディアとしてTVやVTRなどのメディアに勝る可能性を有している。しかし、従来のアニメーションツールでは、その表現力、開発効率、シミュレーション性などに課題があった。今回シミュレーション性を伴うアニメーションを効率良く作成するためのインタラクティブアニメーション記述言語とその処理系「AV-Script」（以下 AV-Script）の開発を行なったので、その機能と実現方法について報告する。

2　アニメーションモデル

現在最もよく使用されているアニメーションモデルは、セルアニメーションをモデル化したフィルムモデルである。（図1）このモデルの利点は、非常に簡単でアニメーションフィルム作成者にとっても理解しやすいということである。

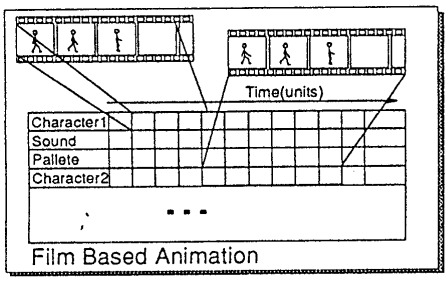

図1　フィルムモデルの概念図

インタラクティブアニメーションを作成する場合、このモデルには以下のような問題点が考えられる。

1．登場人物（以下キャスト）の大きさ、移動速度、移動軌跡等の修正変更をする場合はキャストに対応したフィルムをもう一度作成し直さなければならず、手間がかかる。

2．時間軸上にキャストを配置することによりシナリオを作成するため、衝突等のキャスト同士の相互干渉を表現する場合、修正変更が難しく複雑な相互干渉を表現しようとすると限界がある。

3．インタラクティブ性は基本的にフィルムの切替で実現されるため、時間軸上の単純ブランチのみである。

4．時間制御は絶対時間でのみ行なわれる。

これに対しオブジェクトモデルでは各キャストをオブジェクトとして定義し、その時空間上での振舞いを記述することによりシナリオを作成する。オブジェクトモデルでは上記1．3．に関しては次のように改善されている。

1．各キャストがオブジェクトとしてカプセル化されているため、キャストの属性や振舞いを変更するだけで良く簡単である。

3．インタラクティブ性がオブジェクト間のメッセージパッシングにより記述されているためより複雑な表現が可能である。

しかし、2．4．に関しては次のような問題点がある。

2．修正変更は簡単になっているが、各キャストの状態を統合的に管理する機構がないため、相互干渉の記述が難しい。

4．相対／絶対時間の変換機構がないため記述が難しい。

今回オブジェクトモデルを拡張したモデルを設計した。（図2）今回設計したモデルでは、複数のキャストを並列に実行する機構を実現したことにより、各キャストを自律したオブジェクトとして定義し、実行できるようになった。

通常のオブジェクトモデルの備える複数キャスト間のメッセージパッシングやユーザとの対話の機構の他にオブジェクトの管理機構を拡張することにより、複数キャスト間の相互干渉を簡単に記述変更できる機能を実現している。また、複数時間軸の制御を行なうタイムダイレクタを設けることにより唯一の絶対時間を制御するだけでなくキャスト毎に定義された相対時間を簡単に扱える。

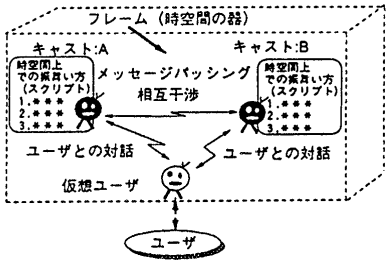

図2　AV-Script のアニメーションモデルの概念図

3　AV-Script システムの構成

AV-Script のシステム構成を図3にあげる。

AV-Script は UNIX ワークステーション、X Window,Motif 上で開発されており、開発言語は C++である。

AV-Script の言語処理系はオブジェクトマネージャ等の機能別に分離された制御ソフトウェア層と、データ及びユーザーインタフェースからなるオブジェクトライブラリ層から構成されている。

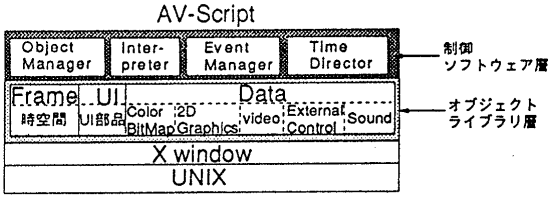

図3　AV-Script のシステム構成

Interactive Animation Scripting Language and its Environment 'AV-Script'
Hiroyuki HAMADA, Kiyomi SAKAMOTO, John ZUROWSKI
Media Research Laboratory, Matsusita Electric Industrial Co., Ltd.

4　AV-Script システムの説明

AV-Script では、素材データを実体データとして持つデータオブジェクトとスクリプトを実体データとして持つスクリプトオブジェクトを統合した自律的なオブジェクトとしてキャストを定義している。各キャストは、それ自身の振舞い方をスクリプトとして自分の中に持っているため、キャストの属性、振舞いの修正は、スクリプトのパラメータを変更するだけで、非常に簡単に修正できる。そしてユーザはスクリプトに各キャストの振舞い方を個別に記述することにより、シナリオを作成する。

AV-Script 言語処理系（図4）により各キャストが制御され、シナリオが展開される。

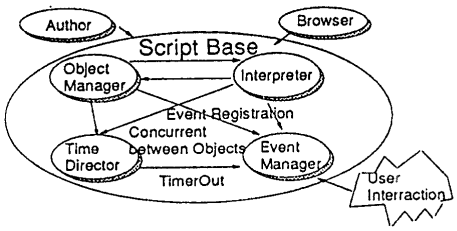

図4　AV-Script 言語処理系の構成

各マネージャの機能、特徴を以下に述べる。

オブジェクトマネージャ

従来、オブジェクトモデルを採用したシステムのオブジェクトマネージャは各オブジェクトの状態変化を管理する機能を持つ。しかし、それは各オブジェクトで閉じており、それらを統合的に管理する機能を持たないため、オブジェクト間の相互干渉の記述は難しい。AV-Script では、この機能を拡張することによって、衝突等のオブジェクト間の相互干渉を簡単に記述することを可能としている。そのメカニズムを以下に示す。

1．ユーザが、必要な状態変化イベントの種類とそのイベントを受信可能なオブジェクトのリストをオブジェクトマネージャの状態変化イベントテーブルに登録する。

2．オブジェクトマネージャがオブジェクトから状態変化の通知を受けとる。

3．オブジェクトマネージャは状態変化イベントテーブルを調べる。

4．状態変化を起こしたオブジェクトが登録されていれば、オブジェクトマネージャはその状態変化に関する相互干渉が成立しているかどうかを調べる。

5．相互干渉が成立していればオブジェクトマネージャはイベントマネージャに、その状態変化イベントを送信する。

イベントマネージャ

AV-Script はインタラクティブ性をメッセージパッシングにより記述している。イベントマネージャはオブジェクトマネージャ、タイムダイレクタ、インタプリタ、ユーザからのメッセージをイベントとして受けとり、各イベントの実行順序、実行単位を制御している。これらの制御により、各イベントの並列実行、複数イベントの同期／非同期実行の制御を行なっている。

インタプリタ

インタプリタは、並列実行が可能なように設計されている。イベントマネージャからスクリプトの実行依頼が来ると、インタプリタは指定されたスクリプトを指定された実行単位だけ実行し、制御をイベントマネージャに戻す。上記のイベントマネージャとインタプリタの機能により、AV-Script では各キャストの並列実行を実現している。

従来のアニメーションツールでは、複数のキャストを同時に移動させようとする場合には、

1．各キャストについて微小時間Δｔにおける移動量Δｓだけキャストを移動させるスクリプトを作成する。

2．1で作成した各スクリプトを順に呼び出して、全キャストを移動させるスクリプトを作成する。

3．2で作成したスクリプトを微小時間Δｔ毎に実行する。

のようにスクリプトを作成する必要があった。この記述方式では、必ず上記2のスクリプトのような全キャストを管理しメッセージを送るスクリプトが必要で、スクリプトをキャスト毎に完全に個別に作成することができないため、キャストの追加、キャストの動作の修正、追加などが非常に難しい。

これに対し AV-Script では複数スクリプトを並列に実行可能なため、それぞれのキャストの動きを記述するスクリプトを個別に作成すれば良く、上記2のスクリプトを作成する必要がない。このためキャストの追加、キャストの動作の修正、追加時に発生するスクリプトの修正箇所が少なく、これらの追加修正が非常に簡単である。

またこのインタプリタはオブジェクト指向言語であり、クラス定義、クラスの継承の機能を備えている。AV-Script ではこの機能を備えているためスクリプトの追加、修正、複雑なクラスの作成が非常に容易である。

タイムダイレクタ

タイムダイレクタはシステム全体の時間管理、タイマイベントの発行、制御を行なう。タイムダイレクタは絶対時間軸の他に複数の相対時間軸を作成、制御することができる。タイムダイレクタはその相対時間軸を絶対時間に変換することにより、相対時間の制御を実現している。

この相対時間軸を各キャストに張り付けることにより、各キャストごとに固有の時間軸を持たせることができる。

5　まとめ

現在、「AV-Script」の制御ソフトウェア層の原型開発を完了している。今回新たに実現したキャスト間の相互干渉、各キャストの並列実行、相対／絶対時間の制御の機能により、自律オブジェクトの表現が可能となり、シミュレーション性のあるアニメーションを簡潔に記述することが可能となった。今後の展望としてオーサリング環境の開発や、３Ｄグラフィックス、アニメーション自動作成機能等のオブジェクトライブラリ層の充実を図りたい。

6　参考文献

1) 阪本．他　『インタラクティブアニメーション記述言語とその処理系「AV-Script」』 第６回情報処理学会情報メディア研究会予稿集（1992 年 5 月）