

## マルチメディア情報のオブジェクト指向分析

今村幸裕 笹平拓哉 橋本英俊 渡邊崇文 三井田 隆 樋口克洋 秋山義博

金沢工業大学 人間・情報・経営系、情報工学研究科

9 2 1 石川県野々市町扇が丘7-1

ハイパー音声とマルチメディアCAIを具体例として取り上げ、マルチメディアアプリケーションシステムの特徴を捉え表現する為のオブジェクト指向分析法の改良を述べる。ここでは、次の特徴に着目した：1) マルチメディア情報のコンテンツに含まれるオブジェクトタイプは目的と分析者によって異なる、2) マルチメディア・オブジェクトに対する操作は並行操作で実現される、3) マルチメディア情報を形成するオブジェクトは同質多数の内部オブジェクトを持つ。これに対しては、分析対象のマルチメディアオブジェクトについて汎化を十分におこなう、マルチメディアオブジェクトに含まれる操作を並行プロセス上で実現しマルチメディア情報を表す、再帰構造と並行メソッドを内蔵した集合オブジェクトタイプの分析法と記述を提案する。

## Object-Oriented Analysis of Multimedia Information

Yukihiro Imamura, Takuya Sasadaira, Hidetoshi Hashimoto, Takafumi Watanabe,  
Takashi Miida, Katsuhiko Higuchi, Yoshihiro Akiyama

Information and Computer Engineering, Kanazawa Institute of Technology  
7-1 Ohgiga-oka Nonoichi-machi Ishikawa 921 Japan  
E-mail: akiyama@infor.kanazawa-it.ac.jp

We describe several improvements on the currently proposed Object-Oriented Analysis techniques. Finding the next characteristics of multimedia information and the processing: 1) Object types may be resulted with different definition according to analysts or application's objectives, 2) Operations on multimedia objects are specified as multiple concurrent processes to realize effects familiar to us in daily life, and 3) Many objects are working collectively and have similar characteristics, it is proposed that the generalization of object types / operations should be made enough to see the "true" view of multimedia objects, the method implementation should be based on concurrent processes with synchronous control, feed-in/out, event driven control etc., and the collection class should be utilized extensively.

## はじめに

この論文では、時間的変化を示す音声などの情報やマルチメディアCAIのコンテンツの構造とその処理システムの構造を求めるオブジェクト指向分析の改良方法の一案を述べる。

マルチメディアシステム設計にオブジェクト指向アプローチを適用する例は多くあるが、これらのほとんどは、動画像、音声などをファイルとして組み込む方法で利用される。代表的な例としてprintコマンドが文字ファイルと静止画像ファイルに対してアプライされる仕組み（ポルモルフィズム）がある。現在のハイパーテキスト概念は、この概念と文字ベースのユーザインターフェース（CUI）とグラフィカルユーザインターフェース（GUI）の混合で、目次、語彙、説明文、画像などの静的構造をノードとしてそれを利用した関連情報の効果的なアクセス方法を実現した。これらのノード間の意味的関連を静的リンク（後述するように、時間的に生成消滅しないと意味で）を張って現し、情報アクセスのナビゲーションを助ける。文字ノード以外であれば、例えば、静止画像の良く解らない一部分を”開く”と、その説明文、拡大静止画像、関連動画像などをポップアップ小ウィンドウに表示したり音声や音楽を鳴らすように出来る。

これに対して、音声や動画像に代表されるようなマルチメディア情報とその処理の特徴は、時間空間的広がりの中で考えなければならない事である。その中でも、単純なマルチメディア情報として、音声情報についてハイパーテキスト風の仕掛けを考える事が出来る。一方、その様なシステムのユーザインターフェースはマルチメディア情報に対する操作になる事から、この設計アプローチは、オブジェクト指向に基づくことが望ましい。同様の事が、マルチメディアCAI情報についても言える。以下では、この音声のハイパー化（ハイパー音声）とマルチメディアCAIコンテンツの分析について、オブジェクト指向技術適用の観点からその改良を述べる。

## ハイパー音声とマルチメディアCAI

音声、音楽、動画像などの時間的変化を持つマルチメディア情報について、ハイパーテキストのリンクのような仕掛け（この時間依存型リンクを動的リンクと以下では呼ぶ）の有効な幾つかのアプリケーションを考える事が出来る<sup>1</sup>。例えば、音声情報で、物語や科学等の話を聞いている時、意味の解らない語彙、表現、文章等が出てきた場合、それを聞いている”その間に”、要求ボタンをクリックすると、その補助的説明情報（テキスト、音声、動画像などの適当な形式）が表示されたら大変助けになる。又、音楽を聴いている場合でも、テーマ、メロディー、解説等に関して知りたくなったら、その時に、関連情報を要求できて聴くことが出来ればいろいろな楽しみ方が出来るであろう。このような性格をもった音声情報をハイパー音声と以下では呼ぶことにする。

先ず、ここで考えるハイパー音声の概略を述べる。ひとつの物語は、本文と本文中で使用されている語彙の幾つかに関する説明文から構成される。本文と説明文は複数の文からなり、一つの文は、簡単な為に、複数のフレーズから成り、それは単純フレーズと説明文に”リンク”をもったハイパーフレーズに分けられるとする。

ひとつの物語をシステムが実行を開始すると、順次に与えられた文のフレーズを実行する。フレーズの実行とは、”その部分の音声データを音声生成ボードに出力し音を出す事である”とする。時間が経過すると、フレーズが次々と実行される。特に、ハイパーフレーズを実行している間、ひとつのボタンを表示して、このボタンを押すことにより関連した説明文を聞く事が出来る。説明文の実行が終わったらこのボタンを消して、ハイパーフレーズの次のフレーズから実行を再開する。ハイパーフレーズの音声流れている間だけそのリンクが有効になる。

マルチメディア音声データは、”フレーム”単位でまとめられたファイルである。再生領域は、開始と終了のフレーム番号で指定され、ユーザはこれらの番号を変える事により好みの部分を再生できる。以下の議論では、図-1に示すように、ひとつの文からなる物語とそれに付随した説明文を考える。これは、実際には音声データファイルとして与えられるので、フレーズは

<sup>1</sup> 人間は、時間軸に対して”現在”に於いてのみアクションを起こす事が出来る。過去や未来に対しては、まったく操作出来ない。

それぞれ開始と終了フレーム番号で指定される。最近のマルチメディアOSはQOSが高いので、この開始・終了フレーム番号の代わりに、ファイルの先頭からの再生時間を変数にして開始・終了時間で指定できる。

次に、マルチメディア技術をCAIに応用する場合を考え、その一つの例として、オペレーティングシステムに関する自習用CAIについて考える。コンピュータハードウェアの動作とOSの内部機能・操作の様子、特に、CPU割り込みと割り込みハンドラー処理系や並行プロセス系は、実際にコンピュータシステムを初めて学習する者にとっては複雑と思われる。これを、アニメーションやシミュレーションを利用して可視化し説明音声を流しながらそのシステムの動きや関連情報を通して理解を助けることは意味がある。コンピュータシステムは並行処理を利用するシステムであるので、この特徴をマルチメディア技術を利用して学習支援CAIを実現する事を考えなければならない。このようなCAIシステムの特徴は、

- ◆実際に何が起きているかを画像と音を同時に使いながら(並行に)その時間的な変化(動画像)を示す、
- ◆CAIの場合には、ハイパーテキスト形式の説明に加えて、ハイパーサウンド、並行にできる種々の効果音、BGM等の利用が重要になる、
- ◆仮想実験環境を提供して、初期設定とシミュレーション機能を利用しながらシステム動作の学習を支援する、
- ◆概念レベルの学習を支援する、

等である。3番目と4番目の項目は一般製品の場合とは異なって教育向けCAIについて重要となる。更に、一つの例として、先ず入出力割り込みの全体シナリオ概略の基本的部分について考える：

- ◆CPU、データと制御のバス、メモリ、入出力制御等を構成要素とするコンピュータハードウェアの基本的構造図と機能を表示・説明する、
- ◆一つのプロセス(プログラム)がファイルレコード読み取り要求を出してい

てその読み込み完了を待っている状況を表示・説明する、

- ◆入出力制御(チャネルやI/Oアダプター等)が行う入出力操作をとCPUの命令実行とが時間的に並行に行なわれている状況をその基本構造図上でアニメーション表示・説明する、
- ◆ひとつのレコードの入出力操作の完了事象は、入出力制御からのCPUに対する割り込み要求、CPUの現命令実行終了時での割り込み検出<sup>2</sup>と割り込み操作(PSWの入れ替え)実行、割り込み処理ルーチンの実行開始と実行中断(割り込まれた)を受けたプログラムの状態保存までの様子を示す、
- ◆割り込み処理ルーチンの入出力操作完了した装置アドレスとその入出力要求を出したプロセスの割り出しを行い、そのプロセスを実行可能にする処理とその順序を示す、
- ◆プロセスの実行再開を示す。

一般に、仮想実験環境を実現をする場合としない場合とでは、CAIの基本構造・機能が異なる。例えば、次のような場合を考えると、OSのアルゴリズムのシミュレーションも含ませなければならない事が解る：

- ◆ファイルレコード入力要求プロセス(プログラム)ライフサイクル実験：入出力プログラムの実行、入出力サービス待ち発生、プログラム実行インターセプション現象等について初期パラメータ設定とその後の実行状況調査
- ◆共有ファイル制御実験：共有ファイルの設定、並行プロセスとクリティカルセクション制御、不完全又は完全なクリティカルセクション定義を持つ並列プログラムの共有ファイルアクセス、性能評価等の実験調査

等が実現される必要があり、このシステム設計は単純ではない。しかし、ここではシステム設計の議論は行わないので省く。

### オブジェクト指向アプローチの適用

<sup>2</sup> 割り込み可能命令などとも言及出来る。

オブジェクト指向アプローチを、以上に述べたマルチメディア情報とその再生システムの構造分析に適用する事考える。オブジェクト指向アプローチは、ビジネスアプリケーションの分析・設計支援を中心に開発されているので、これを改良する必要がある。その注意点は以下の通りである：

A) ハイパー音声の場合：

- ◆ハイパー音声リンクは一般的で良い（語単位に限らなくてもよい）。音声データのどの部分をオブジェクト（フレーズノード）にするかは分析者やユーザにより変わりうる。
- ◆物語、本文、文、フレーズ等のオブジェクトのデータ構造は上位ノードは順序付けられた多数の下位ノード集合として木構造で表され、その実行は再帰的可能である。
- ◆本文オブジェクトのハイパー音声は、全て時間的流れに沿ってシングルスレッドで実行される。ハイパー音声リンクは時間的に出現消滅するように見え、ボタン（ハイパーボタンと呼ぶ）オブジェクトの表示・消滅によって知らせる。従って、プリミティブオブジェクトのオブジェクト操作はシングルスレッドである。
- ◆説明文オブジェクトの実行はユーザがハイパーボタンをクリックすると生成されるトリガーで起動され、終了後、サスペンドされていたオブジェクトの実行が再開される。ここで、説明文の実行も本文実行も同様なので再帰構造が有効であると考えられる。

B) マルチメディアCAIの場合：

- ◆シナリオは階層構造を持ち逐次実行する。ユーザは一時点では一つのシナリオを選択実行出来る。
- ◆シナリオ、効果音、BGM等を夫々オブジェクトとして定義する。シナリオとその下位シナリオ（サブシナリオ）オブジェクトに関する操作は並行プロセス系上で実現する。効果音等のオブジェクトはコンストラクターが作りその実行はイベント駆動、経過時間変数

を用いて制御する<sup>3</sup>。これにより、多様な効果音と動画像等を同期しながら同時並行に処理出来る。

- ◆オブジェクトのライフサイクルは簡単の為に全て階層的であるとする。上位オブジェクトを親オブジェクト、下位オブジェクトを内部オブジェクトと呼ぶと、親オブジェクトが実行終了するときは、その内部オブジェクトは全て実行終了する。

## マルチメディア情報とそのオブジェクト指向分析

オブジェクトスキーマ：以上に述べたマルチメディア情報とそのシステムのオブジェクトのタイプと関連（association）構造を求める分析を考える。注意する特徴は、

- ◆オブジェクト定義は分析者によって異なる可能性がある（“名詞をオブジェクト又はオブジェクトタイプとする”のようなルールは必ずしも適用しなくても良い）、
- ◆マルチメディア・オブジェクトは動的に振る舞うものが多い、
- ◆オブジェクト関連情報は、イベントスキーマと深く関連している事が多い、
- ◆並行プロセスを内蔵したメソッドを利用する、

等である。更に、オブジェクト指向分析方法論は対象システムを出来るだけ孤立系に近い形で選択するように強制する効果があることが解る。つまり、部分的なオブジェクトスキーマではシステム振る舞いを十分に書き下すことが出来ない場合が多い。ここでは、ユーザインターフェース情報を必要分だけ取り込むことでシステム全体の記述が完成する。

ハイパー音声の場合、上に述べた特徴のうち、はじめとその次の特徴に注意しながらオブジェクトスキーマを求める。初めに、動的構造にまったく依らないレベルで、物語とユーザインターフェース関連のオブジェクトを挙げ、それらの分類を行って、次にオブジェクトタイプと汎化・構成・タイプ分割などを適用した関連基本構造図（図一1）を求める。次に、夫々のオブ

<sup>3</sup> QOSを仮定する。

ジェクトを生成する事において関連付けられるリンクを追加する(図-2)。オブジェクトスキーマ分析の最初の段階ではここまで求める事が出来る。マルチメディアCAIの場合、シミュレーション対象シナリオに登場するオブジェクトについてクラシフィケーションを行いオブジェクトタイプを求め、同様の単純化を行う(図-3)。この図では、まだ並行に動作するオブジェクトタイプが示されない。

### イベントスキーマ分析

ハイパーメディアシステムやマルチメディアCAIシステムの場合、ユーザインタラクション駆動とシステム駆動による処理が合併しているため、この両方の見方からシステム処理仕様を求める事を考える。互いに独立で最少の処理単位をトランザクションと呼び、このトランザクション全体集合がシステムに実装したい機能全体を表す。そのトランザクションの一つのインスタンスが一つの具体的処理である。次に、一つのトランザクションの処理仕様はオブジェクトタイプに対する一連の操作として記述され、逐次操作と並行操作を書き下すことが出来る。この一連の決められた操作をイベントスキーマと呼ぶ。一つのトランザクションに対して一つのイベントスキーマが存在する。

ハイパー音声の場合、イベントスキーマの再帰と汎化構造を求めることが重要である。再帰構造はフレーズ単位の実行制御を簡単にしてくれる。操作に対する汎化構造を求めることは、出来上がった操作仕様一般性を高める。図-4は例題テキストの実行に関する個々のイベント流れ図であるが、図-5が汎化と再帰構造を押さえた表現である。

各トランザクションについてイベントスキーマを求めると、その処理に関するオブジェクトタイプ間の(真に動的な)関連をオブジェクトスキーマ上に追加出来る(図-6)。このように、3段階に分けて分析して最後にそれらを合わせた図を作成すると、オブジェクトスキーマが得られることが解る(図-7)。これから、ユーザインターフェスの必要部分(夫々の目的にあったボタンの定義等)がオブジェクトスキーマに追加されリンクされる事が解る。

マルチメディアCAIの場合、通常のイベントスキーマでは十分ではない。図-8は、“レジスターを待避する”を詳細化したものであるが、真ん中の2つの操作は時間的に同時実行されなければならない。更に、図-9は“割込み操作の説明”に対する段階的詳細化を行った通常のイベントスキーマ図であるが、これは、6つの部分に分けて逐次実行する事を示している。しかしながら、CAIの場合、これでは十分ではなく、図-10に示すような複数内部オブジェクトとそれらの並行操作が解るような表現を用いて分析する必要がある。このような図表をPTチャート(パラレルタイムチャート)と呼び、横軸が経過時間、縦軸が並行オブジェクトとその操作を表す。

### オブジェクトの状態遷移表現

オブジェクトタイプごとの状態遷移図は、オブジェクトのライフサイクルを表現する。オブジェクトの状態間遷移はイベントスキーマにあらわれる操作により引き起こされる。この意味で、この両者はコンシステントでなければならない。つまり、再帰化・汎化を施したイベントスキーマとコンシステントでなければならないことから、ハイパーサウンドの場合の状態遷移図は非常に簡単になる(図-11)。マルチメディアCAIの場合も同様のことが言える。

Use Caseでは、インタラクションダイアグラム、状態遷移図、交換メッセージを併せて使用表示する方法であるが、逐次処理表現にその重点があり、マルチメディア情報オブジェクトの表現には改良が要る。

### 分析・設計とプログラミングの統合

図-12にOMT法とJ. Martin法に対するC++言語機能の対応表を示す。これから、オブジェクト指向分析・設計からオブジェクト指向プログラミングへの接続は必ずしも良くないことが解る。更に、今までに述べたハイパー音声とマルチメディアCAIシステムのオブジェクトスキーマやイベントスキーマ等をC++プログラムに変換出来るかどうかは自明ではない。以下では、この問題について特に重要な部分について述べる。

ハイパー音声の場合、特徴的な事は物語本文や説明文が順序付き文から成り、一つの文は順序

付きフレーズから成っている事である。このようなオブジェクトタイプを図-1.3のように表しC++プログラムとしては順次コレクションクラスを割り当てられる。又、C++テンプレート機能はパラメータで定義出来るオブジェクトタイプの集合に対して共通の一つの定義を対応付ける事が出来る。

マルチメディアC A Iの場合、並行プロセスを内包した操作の実現方式を与えなければならない。図-1.1に示すように、内蔵オブジェクトに対する操作起動のタイミング制御は複雑になるが、イベント駆動、フェードイン・フェードアウト機能、経過時間による起動、等の制御法利用を考えると良い。又、共通表示画面に対する並行更新のイメージ表現は時間的インターリーブを行う事によって現される。更に、内部オブジェクトは、クラス内クラスとしてコード化出来る。

### まとめ

マルチメディア情報のオブジェクト指向分析について述べた。ここで解った特徴的な事は、1) マルチメディア情報のコンテンツ内に入ると、オブジェクトは必ずしもいつもはっきり決められるとは限らない、2) オブジェクト操作は並行プロセス上で実現しなければならない、3) 同質多数のオブジェクトを取り扱う必要性から、集合クラス利用が必要になる、等である。現在版オブジェクト指向分析技術・方法論に対するこれらの改良は、マルチメディア情報とその処理を考える上で必要であると言える。

### References

J. Martin, Principles of Object-Oriented Analysis and Design, Prentice-Hall International Inc., 1993

J. Martin, J. Odell, Object-Oriented Methods: A Foundation, Prentice-Hall Englewood Cliffs, NJ, 1995

J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorenson, Object-Oriented Modeling and Design, Prentice-Hall International Inc., 1991

P. Coad, E. Yourdon, Object-Oriented Analysis, 2-nd Edition, Prentice-Hall International Inc., 1991

山城明宏, et. al.; 「画像検索システムへのOMT手法の適用」、情報処理学会研究報告 Vol. 84, No. 3, 1992

飯島正, et. al., 「画像ファイリングシステム開発へのオブジェクト指向方法論の適用事例」情報処理学会研究報告 Vol. 85, No. 3, 1993

宮本衛市, et. al., 「オブジェクトのモジュール化による複合オブジェクトの生成」情報処理学会研究報告 Vol. 87, No. 3, 1992

Roger C. Schank, Active Learning through Multimedia, IEEE Multimedia Spring 1994, Page 69

S. W. Samolia, et. al., Content-Based Video Indexing and Retrieval, IEEE Multimedia, Summer 1994, Page 62

D. Moskoowitz et. al., OS/2 Warp Unleashed, Deluxe Edition, SAMS Publishing 1995

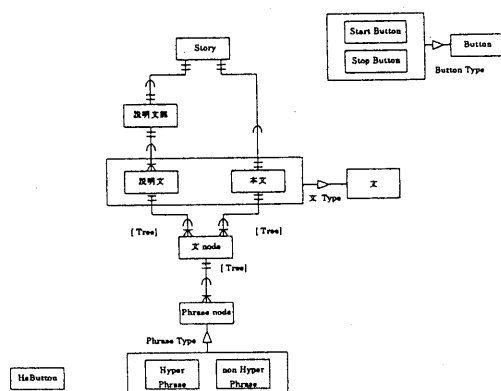


図-1: Structural Object Diagram

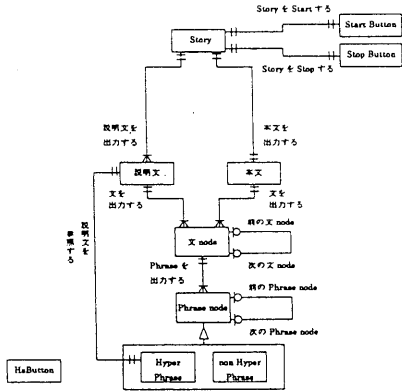


図-2 : Semantical Object Diagram

オブジェクト関連図

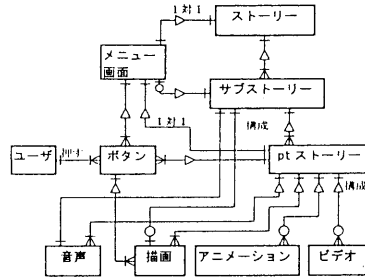


図-3 : MM Object Schema

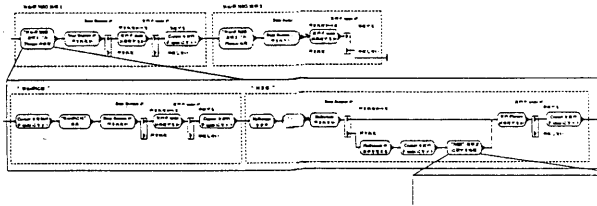


図-4 : HSG OO Event Diagram 具体例

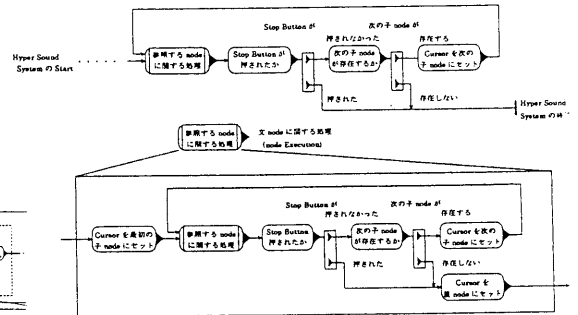


図-5 : HSG OO Event Diagram - 汎化版

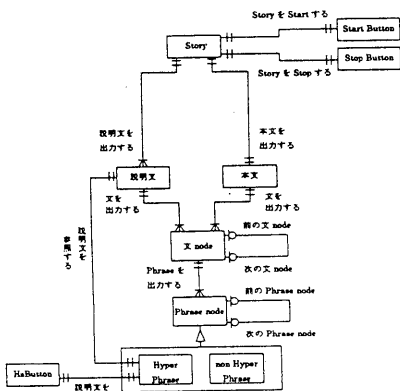


図-6 : HSG OO Schema w/Button links

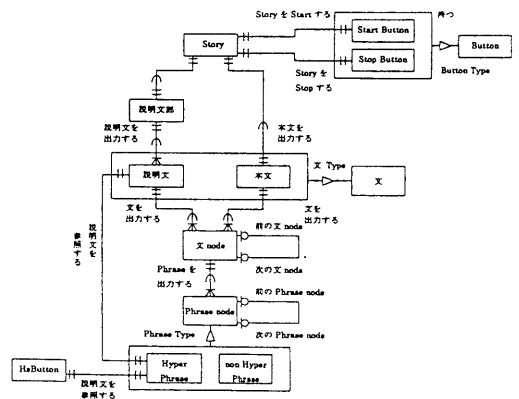
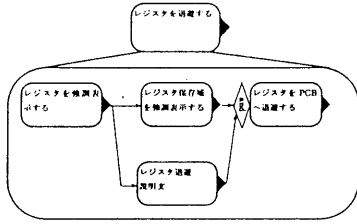
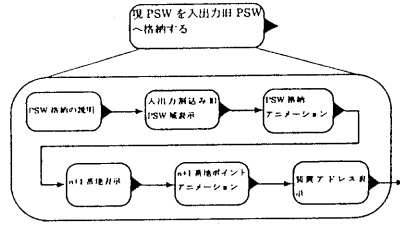


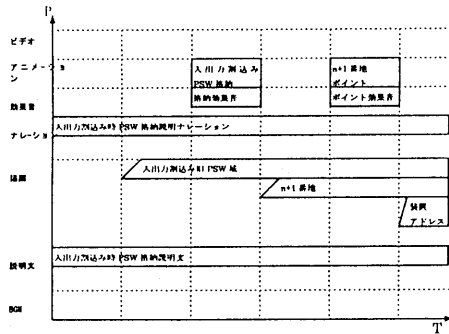
図-7 : 図-1, 2, 6の総合図



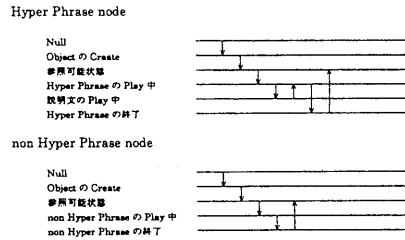
図一 8 : “レジスタを待避する”Event図



図一 9 : “現PSWを入出力旧PSWへ格納する”Event図



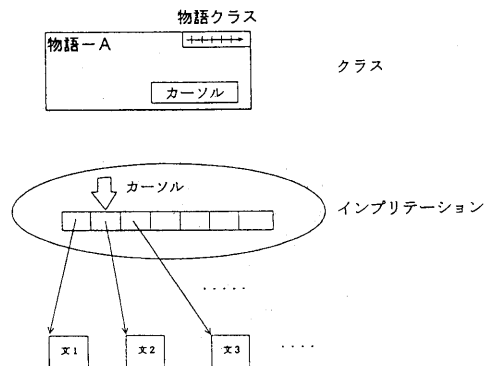
図一 10 : PTチャート



図一 11 : ハイパー音声オブジェクト状態遷移図 (例)

J.Martin	C++	OMT
オブジェクトタイプ	クラス	クラス
関連タイプ	ポインター	関連
イベントタイプ	-	事象クラス
制御条件	-	-
トリガールール	メンバー関数の実行に必要な引数	-
-	コレクションクラス	-
オペレーションの段階的詳細化	関数からの関数呼び出し	伝搬
-	テンプレート	-
.....	.....	.....

図一 12 : J. Martin法, OMT法, C++比較



図一 13 : 順序コレクションクラス図表現