

# オブジェクト駆動の実行環境に関する一提案

山本圭一 吉田 聡 大原茂之  
東海大学

40-10

## 1. はじめに

ソフトウェア設計技法の1つとして、オブジェクト指向では困難であったオブジェクトの動的な生成や消滅をモデル化することが可能なオブジェクト駆動図という記述体系<sup>1)</sup>がある。本報告では、オブジェクト駆動図における移動オブジェクトの受け渡し、メッセージの受け渡し、および各々のオブジェクトのCPUへの割り当てを行うオブジェクト駆動実行システム(Object-Driven Execution System:ODES)を提案する。

## 2. ODES の概要

### 2.1 ODES の入出力

オブジェクト駆動実行システムは任意のオブジェクト駆動図(Object-Driven Diagram:ODD)を入力すると、それに対応する実行結果を出力する。実行結果とは、ODDからの出力オブジェクト、およびODDからの出力メッセージである。図1にODESの入出力を示す。

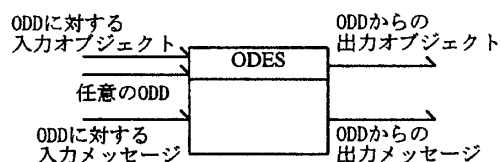


図1. ODES の入出力

### 2.2 ODES の構成

ODESはオブジェクト単位でCPUの割り当てを行う。入力されたODDをオブジェクト単位で実行させるためにアークを切断する。切断したオブジェクトのアークの接続情報を記録し、メッセージの送受信、オブジェクトの受け渡しを行うオブジェクトをトランスポータと呼ぶ。トランスポータは、内部にオブジェクトを保持する。アーク接続情報とは、トランスポータ内部のオブジェクトが、どのオブジェクトとアークでつながっていたかを示す情報である。アーク接続情報は、オブジェクトの移動元オブジェクト名、オブジェクトの移動先オブジェクト名、メッセージの送信元オブジェクト名、メッセージの受信先オブジェクト名で構成される。図2にトランスポータの例を示す。

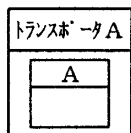


図2. トランスポータの例

ODESにODDを入力すると、図3に示すような分割を行ない、トランスポータを合成する。図3に示すように、内部にバッファオブジェクトAを保持するトランスポータの名前をAとする。バッファオブジェクトBを保持する場合も同様である。バッファオブジェクトと移動オ

ブジェクトを分割したとき、CがAに保持されている場合のトランスポータの名前をA(C)とする。CがBに保持されていない場合のトランスポータの名前をB( )とする。また、図3において、トランスポータA(C)を移動元トランスポータ、トランスポータB( )を移動先トランスポータと呼ぶ。

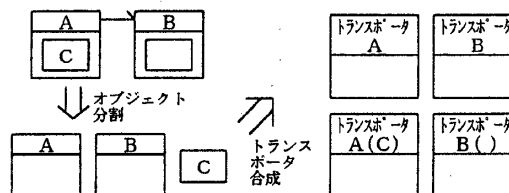


図3. オブジェクトの分割とトランスポータ

ODESはトランスポータ、ODD分割、トランスポータ合成、トランスポータ生成、実行、交信元、交信先、実行待ちの8種類のオブジェクトから構成される。図4にODESの構成を示す。

トランスポータは、内部にオブジェクトを保持し、メッセージの送受信、移動オブジェクトの受け渡しを行う。

ODD分割は、入力されたODDに対して、図3に示すような分割を行なう。

トランスポータ合成は、分割されたオブジェクトとトランスポータを合成する。このとき、各々のトランスポータにアーク接続情報を持たせる。

トランスポータ生成は、常にトランスポータを生成する。

実行は、内部のトランスポータが保持しているオブジェクトに対して、CPUの割り当てを行う。

交信元は、バッファオブジェクトが内部に保持する移動オブジェクトを移動させる場合、移動元のトランスポータを保持し、移動オブジェクトの送り出しを行なう。

交信先は、移動オブジェクトが移動する場合、移動先のトランスポータを保持し、移動オブジェクトの受け入れを行なう。オブジェクトがメッセージを送信する場合も同様に、送信先のトランスポータを保持し、メッセージの受け入れを行なう。また、図4において、交信先に対するアークは、交信先が入力されたオブジェクト、メッセージを一時的に保管する入力バッファを持つことを示す。

実行待ちは、CPUを割り当てられていないオブジェクトのトランスポータを保持する。また、トランスポータの実行への移動を管理する。実行待ちはトランスポータを有限個保持することができる。

### 2.3 ODES の動作

次に、ODESの動作例を示す。

- (1) ODESに任意のODDが入力された場合  
任意のODDを受信したODD分割は、ODDを図3

A proposal on execution environment of Object-Driven.  
Keiichi YAMAMOTO, Satoshi YOSHIDA,  
Shigeyuki OHARA  
Tokai University.

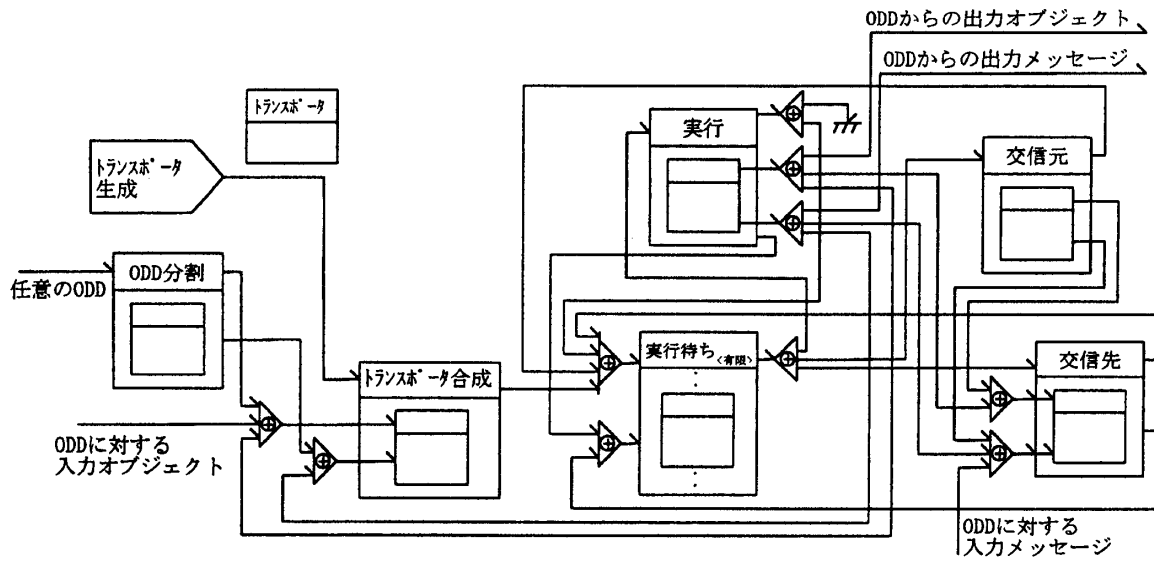


図4. ODESの構成

に示すように分割し、分割したオブジェクトをトランスポート合成内部のトランスポートに移動させる。さらに、トランスポート合成内部のトランスポートにアーク接続情報メッセージを送信する。合成されたトランスポートは実行待ちに移動する。

(2) オブジェクトが生成される場合

実行中のオブジェクトがオブジェクトを新たに生成した場合、生成されたオブジェクトをトランスポート合成内部のトランスポートに移動させる。さらに、トランスポート合成内部のトランスポートにアーク接続情報メッセージを送信する。

(3) 移動オブジェクトが移動する場合

実行中の移動オブジェクトが、他のバッファオブジェクトに移動する場合、移動オブジェクトを保持しているトランスポートは送信先に移動オブジェクトを移動させる。送信先は移動オブジェクトを入力バッファに格納し、実行待ちに対して、移動先呼出メッセージを送信する。移動先呼出メッセージを受信した実行待ちは、送信先に移動先のトランスポートを移動させる。送信先オブジェクトは入力バッファにある移動オブジェクトを移動先のトランスポートに渡す。移動オブジェクトを受けとったトランスポートは実行待ちに移動する。

(4) オブジェクトがメッセージ送信する場合

実行中のオブジェクトがメッセージを送信する場合、送信先にメッセージを送信する。送信先は、入力バッファにメッセージを格納し、実行待ちに対して、送信先呼出メッセージを送信する。送信先呼出メッセージを受信した実行待ちは送信先にメッセージ送信先のトランスポートを移動させる。送信先オブジェクトは入力バッファにあるメッセージを送信先のトランスポートに渡す。メッセージを受信したトランスポートは実行待ちに移動する。

(5) バッファオブジェクトが移動オブジェクトを移動させる場合

実行中のバッファオブジェクトが、内部の移動オブジェクトを他のオブジェクトに移動させる場合、実行中のバッファオブジェクトは実行待ちに対して、移動元・移動先呼出メッセージを送信する。移動元・移動先呼出メッセージを受信した実行待ちは、移動オブジェクトを保持しているトランスポートを送信元に、移動先のトランスポートを送信先に移動させる。送信元に移動したトランスポートは内部の移動オブジェクトを送信先のトランスポートに送り出す。

3. おわりに

本報告では、オブジェクト駆動における実行環境を提案した。実行環境を開発することによって、オブジェクト駆動図を実行させることが可能となった。

今後は、本研究の成果に基づいてオブジェクト駆動の実行環境をインプリメントする予定である。

参考文献

- 1) 吉田, 大原: オブジェクト駆動によるオブジェクト指向プログラミングの効率化, 情報処理学会サマワークショップ・イン・立山論文集, pp.41-48(1995-7).

付録

・オブジェクト駆動図の記述体系

図 a にオブジェクト駆動図の記述体系を示す。

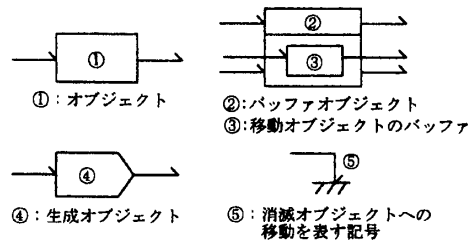


図 a. オブジェクト駆動図の記述体系

図 a において、移動オブジェクトの存在していることを示す場合は、③のバッファの中に移動オブジェクト名を記述する。