

オブジェクト指向分散環境OZ++の開発環境の構想

2 T-3

籠 浩昭\*(三菱総合研究所) 吉田 泰光\*(日本ユニシス)

音川 英之\*(シャープ) 新部 裕\*(三菱総合研究所)

塚本 享治(電子技術総合研究所)

\* 情報処理振興事業協会「開放型基盤ソフトウェア研究開発評価事業」研究員

1 はじめに

オブジェクト指向分散環境OZ++は、オブジェクトの共有と交換の機構を利用して、ネットワーク接続された計算機群上で動作する分散アプリケーションの開発を容易にすることを旨としたシステムである[1]。

ここでは、OZ++における分散処理アプリケーションの開発を支援する開発環境の構想と、現在構築中の開発支援ツールの概要に関して述べる。

OZ++開発環境では、各プログラマ個人々の開発作業を支援することを目的としている。

2 OZ++の特徴

OZ++の特徴には以下のようなものがある。

- 広域に計算資源が分散した環境で動作する管理系を持っている[2]。
- クラスの再利用を促すためにインタフェースの属性に基づくクラス版管理を行っている[4]。
- オブジェクトは各計算機上に分散した実行系で並列に実行される。

OZ++環境は、UNIX等の既存環境と共存しており、相互の環境の機能利用が容易に実現できることを目指している。

3 OZ++における開発イメージ

OZ++におけるプログラム開発および実行のイメージを以下に述べる。

3.1 分散透過なクラスの利用

プログラムを記述する際には、広域のネットワークに分散したクラスのソースコードやインタフェースの情報を参照または利用する必要が出てくる。

Design Concept of Programming Environment for OZ++:  
 an Object-oriented Distributed Systems Environment  
 Hiroaki Kago\* (Mitsubishi Research Institute, Inc.),  
 Yasumitsu Yoshida\* (Nihon Unisys, Ltd.),  
 Hideyuki Otokawa\* (Sharp Corporation),  
 Yutaka Niibe\* (Mitsubishi Research Institute, Inc.),  
 Michiharu Tsukamoto (Electrotechnical Laboratory)

\*: Research fellow of Open Fundamental Software Technology Project in Information-Technology Promotion Agency, Japan

OZ++開発環境では、ユーザが分散を意識したアクセス方法を準備しなくても、実行系や管理系の機能を利用したブラウザやデバッガ等により自動的に参照または利用できる。

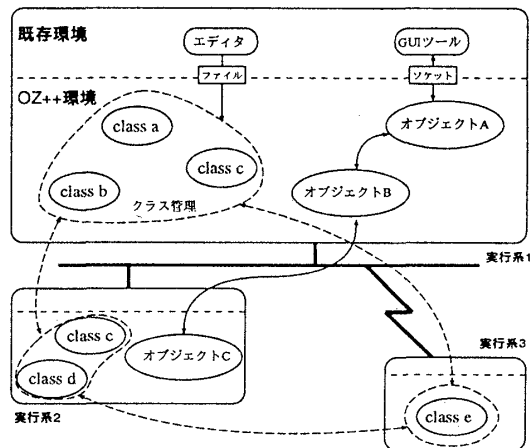


図 1: OZ++における開発イメージ

3.2 既存環境との共存

新しい環境が提供された場合でも、いくつかのツールに関しては使い慣れた既存環境のものを利用したいというユーザの要望は大きいと考えられる。

OZ++では、ファイルやソケット等のインタフェースをOZ++言語で規定することで既存環境との共存が可能になるため、既存環境のツールを併用した開発も容易に実現できる。

例えば、既存環境のエディタを利用してOZ++言語でクラスを記述し、OZ++環境のコンパイラフロントエンドを利用してコンパイルし、OZ++のクラス管理システムに登録することができる。

また、OZ++開発環境におけるブラウザやデバッガ/トレーサのようなアプリケーションは、既存環境のGUIツールを利用してグラフィカルなユーザインタフェースを提供することができる。

### 3.3 分散した計算資源の利用

広域のネットワークで接続された計算機環境では、実行可能コードが分散している場合が多い。オブジェクトの実行は、分散した実行可能コードを組み合わせるにより実現される。

また、効率的にオブジェクトを実行するためには、分散して並列に実行されることが望まれる。

OZ++開発環境では、分散して実行されているオブジェクトの状態を把握し、関連する実行可能コードやソースコード等の情報をネットワーク中から集めてきてユーザに提供することができる。さらに、実際にオブジェクトが実行されている実行系以外から、該当するオブジェクトを対象とした開発作業を行うことができる。

## 4 OZ++開発環境の概要

OZ++開発環境で提供する機能としては以下のようなものがある。

### 4.1 ブラウズ機能

分散管理されるクラスの継承や外部参照の関係、インタフェース、ソースコード、クラスの版に関する情報を、ユーザに位置を意識させることなく提供する。

ツールは、OZ++のオブジェクトとして実装される。特定クラスの詳細な情報を表示するクラスブラウザ、クラス間の関係を表示するクラス階層ブラウザ、ソースコード中の名前とクラスのIDとの対応を表すスクール[3]の情報を表示するスクールブラウザ等、複数のブラウザの組み合わせで実現される。

### 4.2 デバッグおよびトレース機能

広域ネットワーク上に分散し、並列に実行されるオブジェクトに関して、インスタンス変数等のインスペクト機能と実行のトレース機能を提供する。

ツールは、実行系のランタイムルーチンとして提供されるデバッグマネージャと、OZ++のオブジェクトとして実現されるデバッグオブジェクトの組み合わせで構成される。

### 4.3 ランチャ

ユーザがオブジェクトを起動する機能を提供する。

ランチャから起動されたオブジェクトは、OZ++におけるグローバルオブジェクトの永続化の仕組みを利用して保存され、要求に応じて再起動することができる。

ランチャでは、ユーザが起動できるオブジェクトは、特定の抽象クラス(Launchable)を継承している必要がある。

### 4.4 コンパイラフロントエンド

OZ++言語のコンパイルを行うために、ソースコードとスクールとを取り扱い、ユーザとコンパイラとの仲介を行う。

### 4.5 クラス管理

クラス管理システムは、コンパイラフロントエンドからの要求に基づきコンパイラの起動、インタフェースによる版管理、実行可能コードの管理、コンパイル時に利用されたバージョンテーブルの管理などを行う。

また、複製やトランザクションなど提供機能毎にまとめられたクラスライブラリに関してもクラス管理システムを利用して提供する。

## 5 まとめ

これらのツールを利用することにより、OZ++環境の特徴であるネットワーク上に広域に分散したソフトウェアおよび実行資源を利用するアプリケーションの開発や運用が容易になる。

現在、上述のツールの設計・開発を行っている。

また、複数人のユーザが協力しあって作業を行う場合の支援を目的とした協調作業環境に関しては、アプリケーションの検討を行っている。

本研究において熱心な討論を頂いた、藤野 晃延、吉屋 英二(富士ゼロックス情報システム)、千葉 滋(東京大学理学部)各氏に感謝する。

本研究は、情報処理振興事業協会「開放型基盤ソフトウェア研究開発評価事業」の一環として行われたものである。

## 参考文献

- [1] 塚本他: 「オブジェクト指向分散環境 OZ++ の基本設計」, SWoPP '93, Aug. 1993
- [2] 籠他: 「オブジェクト指向分散環境 OZ++ のオブジェクト管理機構の概要」, 情報処理学会第47回全国大会, Oct. 1993
- [3] 吉屋他: 「オブジェクト指向分散環境 OZ++ のクラス管理方式」, 情報処理学会第47回全国大会, Oct. 1993
- [4] 新部他: 「オブジェクト指向分散環境 OZ++ コンパイラによるクラスの版管理」, SWoPP '94, Jul. 1994