

オブジェクト指向分散環境OZ++の 通信機構の基本設計

5F-1

濱崎陽一(電総研)、吉屋英二(富士ゼロックス情報システム*)、
新部裕(三菱総研*)、大西雅夫(東洋情報システム*)、籠浩昭(三菱総研*)、
佐藤啓太(富士ゼロックス情報システム*)、塚本享治(電総研)

1.はじめに

OZ++は、オブジェクトの交換と共有に基づく分散処理環境である。分散システム全体で統一的に管理されたオブジェクトを情報の単位として用いることにより、ネットワークで接続された計算機群の上で容易にアプリケーションを稼働させることが出来る分散システムの開発基盤を目指して、現在その設計を進めている。オブジェクトの交換と共有などの概念は、電総研で開発された分散システムOZ+を基本としており、OZ+で得られた知見をOZ++の設計に反映させる。

ここでは、OZ++の通信機構の基本設計の概要について述べる。

2.システムの構成

OZ++の当初の開発プラットフォームには、ネットワークで接続されたUNIXワークステーションを用いる予定であり、以下それを想定して話を進める。プラットフォームは、次のような階層を持つ(図1)。

ワールド(World): OZ++のシステム全体を指す。ワールドを、内部でbroadcastが行える程度に小規模なサイト単位に分割することにより、システムの規模の大小に対応する。ワールドは、1つ以上のサイトから成る。

サイト(Site): ほぼ等しいコストでステーション間相互の通信が可能な論理的な管理範囲。1つ以上のステーションから成る。サイト内のステーションと他のサイトのステーションとの通信を中継する、ゲートウェイとなるステーションを含む。

ステーション(Station): 1つの計算機(ワークステーション)。1つのステーションが複数のサイトに属することはない。

各ステーションには、ニュークリアスとエグゼキュータと呼ぶ2種類のプロセスが実装される。

ニュークリアス(Nucleus): エグゼキュータを生成・削除したり、通信を管理するなどの機能をもつプロセス。各ステーションで必ず1つ動いている。

エグゼキュータ(Executor): オブジェクトの生成・実行を行うプロセス。一つのエグゼキュータで、複数のオブジェクトが実行され得る。個々のオブジェクトのメソッドの実行はスレッドによって行われる[2]。

OZ++では、名称管理やオブジェクト管理などの

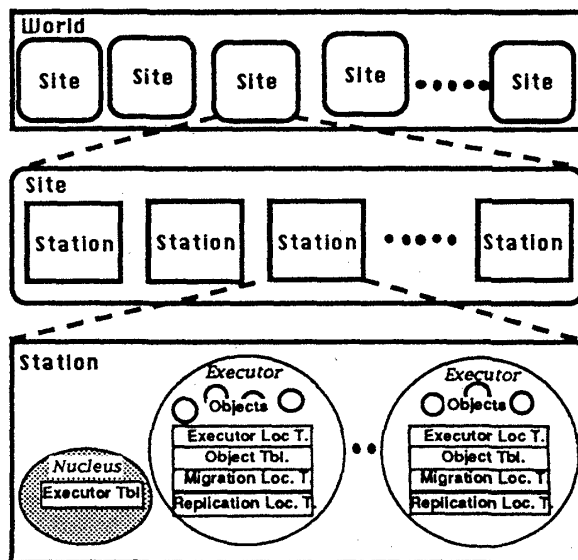


図1 OZ++のシステム構成

管理機構もオブジェクトとして実装され、オブジェクト間のメッセージの交換により、実行が進んで行く。

3.オブジェクトのアイデンティファイア

オブジェクトの名前などの名称は、言語レベル以上で解決され、通信レベルではワールドでユニークなアイデンティファイア(OID)によってオブジェクトを識別する。

サイトやエグゼキュータもそれぞれを識別するためのIDを持ち、ワールドでユニークである。エグゼキュータのIDの一部には、サイトのIDを含み、サイト内のエグゼキュータか、サイト外のエグゼキュータかが判断できる。

OIDは、そのオブジェクトが生成されたエグゼキュータのIDと、そのエグゼキュータで付けられるシリアル番号から成り、オブジェクトの生成から消滅まで変わることがない。エグゼキュータのIDは、ワールドでユニークであるから、エグゼキュータでユニークにシリアル番号を付ければ、各エグゼキュータでローカルにOIDを付与することが出来る。

4.オブジェクト間の通信

オブジェクト間で受け渡されるメッセージもオブジェクトである。オブジェクトはその内部で別のオ

*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

Fundamental Design of Communication Mechanisms of OZ++ : An Object-Oriented Distributed Environment.
Yoichi Hamazaki (Electrotechnical Laboratory), Eiji Yoshiya (Fuji Xerox Information Systems, Co.,Ltd.*),
Yutaka Niibe (Mitsubishi Research Institute*), Masao Onishi (Toyo Information Systems, Co., Ltd.*),
Hiroaki Kago (Mitsubishi Research Institute*), Keita Sato (Fuji Xerox Information Systems, Co.,Ltd.*) and
Michiharu Tsukamoto (Electrotechnical Laboratory)

[* : Researchers of Research, Development and Evaluation of Open Fundamental Software Technology In Information-Technology Promotion Agency, Japan]

プロジェクトを参照しており、複雑な参照関係を持っているのが普通である。そうしたオブジェクトを通信するためには、オブジェクト間の関係を保存したままで、ビット列に変換/逆変換するエンコーダ/デコーダを用いる。エンコーダ/デコーダの方式はOZ+で開発されたオブジェクト間の参照関係をメッセージ内でのみ有効なインデックスに付け換えて、探索順に詰めてゆく1パス方式を改良して用いる予定であるが、その詳細は、言語処理系の設計と並行して決められる。

オブジェクト間の通信は、そのオブジェクトの存在位置によって、つぎの3つの場合がある。

- 1) 同じエグゼキュータ内
 - 2) 同じステーション内の異なるエグゼキュータ間
 - 3) 異なるステーションのエグゼキュータ間
- 1)の場合には、スレッド間の通信で行われる。メモリ空間を共有しているため、エンコーダ/デコーダは簡略化が可能であり、エグゼキュータの機能として実現される。2)の場合には、共有メモリなどのプロセス間通信が用いられ、3)の場合にはネットワークを介した通信が行われる。

5. 通信機構

通信機構は、異なるエグゼキュータ間の通信を司る。ステーション間の通信のうちトランスポート層以下の部分については、移植性の観点から既存のプロトコルを利用する。

エグゼキュータは、そのステーションのニュークリアスおよびサイト内の他のエグゼキュータと直接通信することができる。また、ニュークリアスは、そのステーションのエグゼキュータおよびサイト内の他のニュークリアスと通信することが出来る。

ニュークリアスは、エグゼキュータの通信アドレスと、エグゼキュータのIDの対応表である、エグゼキュータ表(Executor Table)を持っており、ステーション内でエグゼキュータを生成、消滅したときと、他のニュークリアスと相互に通信して情報を得たときに内容を更新する。エグゼキュータが生成される時に、ニュークリアスの持つエグゼキュータ表の内容をもらって、エグゼキュータ局所表(Executor Local Table)の最初の内容とする。通信アドレスを知らないエグゼキュータへ通信する必要が生じた場合には、ニュークリアスの問い合わせ、その内容はエグゼキュータ局所表に書き加える。

また、エグゼキュータは自分で持っているオブジェクトを記録したオブジェクト表を持っている。

6. マイグレーションとレプリケーション

OZ++のオブジェクトは、それを実行しているエグゼキュータから他のエグゼキュータにマイグレートできる。

マイグレーションが起きると、マイグレーションを管理するマイグレーションサーバに、マイグレーションしたオブジェクトのOID、現在のエグゼキュータのIDと何度目のマイグレーションであるかを示すホップ数が記録される。

また、エグゼキュータは、マイグレートしたオブジェクトの位置を記録するマイグレーションキャッシュ表(Migration Cache Table)を持っており、送信を行うごとにキャッシュ表の参照を行なって、テー

ブルにない場合にはマイグレーションしていないものと判断してOIDに含まれるエグゼキュータIDを用いて送信する。自分が持っていないオブジェクトへの通信を受信したエグゼキュータは、キャッシュ表を参照して、もしもそこにあれば、新たな所在を、なければ所在不明を受信失敗のメッセージに含めて返す。送信失敗のメッセージを返されたエグゼキュータは、教えられたアドレスに再送するか、マイグレーションサーバへ問い合わせる。

また、OZ++のオブジェクトは、レプリケートできる。レプリカは、それぞれ異なるOIDを持ち、レプリカの一覧表や、アクセスすべきレプリカの指示などは、トレーディングサーバが管理し、サービスする。

エラーが生じなければ、相手のオブジェクトがレプリケートされているかどうかは、通信に影響しない。エラーが生じた時に、レプリケートされていない、別のレプリカからサービスを受けられるかも知れないので、トレーディングサーバに別のレプリカのOIDを要求する。レプリケートされているオブジェクトを記録しておくために、エグゼキュータは、レプリケーション局所表(Replication Local Table)を持つ。

7. ニュクリアス

ニュークリアスは、エグゼキュータの生成、消滅を行い、またサイト内のエグゼキュータの状態を監視する。

定期的に他のニュークリアスと通信することで、ステーションの状態をある程度把握できる。例えば、定時報告が途絶えたステーションのエグゼキュータへの通信が失敗した場合には、ステーションの障害が推測される。こうした情報は、レプリケートされたオブジェクトの管理など、システム管理以外にも、用いられる。

8. まとめ

OZ++の通信機構について、その基本設計について述べた。OZ+を基礎として、1)集中型の通信モジュールの機能をエグゼキュータに分散することにより通信のオーバヘッドを削減、2)既存のプロトコルを使うことによる移植性の向上、3)マイグレーションのための機構を簡略化、4)レプリケーションへの対応などの改良を図っている。

今後、言語処理系の設計と並行して、通信機構の設計を進め、言語処理系と併に実装を行う予定である。

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、濱崎他：「オブジェクト指向開放型分散システムOZ+の研究開発」、電総研彙報、vol.56, No.9, Sep.1992
- [2] 西岡他：「オブジェクト指向分散環境OZ++の言語の基本設計」、情報処理学会第46回全国大会、Mar.1993
- [3] 濱崎、塚本：「OZ+のオブジェクト通信機構の評価」、情報処理学会第45回全国大会、Oct.1992