

分散アプリケーション開発環境 DIABASE の通信機構の実現*

4W-1

芳竹 宣裕 宮本 剛 上道 悟

日本電気(株) インタフェース技術研究所

1 はじめに

近年、分散コンピューティング環境が整備され、その環境上で動作する分散アプリケーション(分散AP)もその数を増やしつつある。筆者らはこれら分散APを効率よく開発するための支援環境 DIABASE(Distributed application builder's assistance environment)[1]の構築を行っている。

DIABASEの機能のうち、分散APの通信に用いる通信機構を検討したので、本稿ではその実現方法について報告する。

2 DIABASEの概要

DIABASEは、ネットワーク上に分散したサービス実体が連携動作する分散APのコーディング、テストおよびデバッグを支援する開発環境である。特に分散APの通信の必然性に注目し、通信に関わる部分の開発工数の軽減を目指している。

DIABASEは四つのコンポーネントから構成される。

- **通信機構**
分散AP内の通信に使われる機構である。開発者には分散コンピューティング環境提供のAPIを元により抽象度の高いAPIとして提供する。
- **ビルド機構**
コーディングにおいて通信部分の生成を支援する。
- **モニタ機構**
テストフェーズにおいて分散APの動作をモニタし、動作異常を起こした(バグが潜んだ)サービス実体を発見するのに利用される。
- **デバッグ機構**
モニタ機構で発見された動作異常のサービス実体のデバッグを行う。

3 通信機構実現のための基本方針

DIABASEで提供する通信機構は分散APの開発効率を向上させるために以下の点を留意して設計した。

- **サービス部分と通信部分の分離**
各コンピュータ上で提供するサービス処理するソースコードと、コンピュータ相互でデータをやり取り

する通信部分のソースコードとをできる限り分離する。これにより開発者はサービス処理のコーディングへ集中できる。分離された通信部分のコーディングはビルド機構により生成支援させることとする。

- **通信プロトコルの独立**
元来、各通信プロトコルは独自のAPIを持つために、プロトコルを変更すれば通信に関わるソースコードの変更も余儀なくされた。従って、通信の汎用性を持たせる目的で、開発には通信プロトコルに依存しないAPIを用いるのが望ましい。
- **部品化/カプセル化**
通信に関するプリミティブな機能を部品化/カプセル化し、カプセル間の連携で通信路の設定/切断、データ送受信等が行えるようにする。これは、様々な通信プロトコルを扱う上で部品を取り替えるだけで、適当な通信機構が設定できるのは非常に有用と考えるからである。
- **ネイティブなOS、プログラミング言語**
OSおよびプログラミング言語の仕様に改造を加えずに通信機構を実現する。

4 実現方法

基本方針に沿った実現の一方法として、各サービスをデータとその操作手段の集まりと考えると、オブジェクト(サービスオブジェクト)の形でまとめる。通信機構はサービスオブジェクト間の通信を司る(図1参照)。あるサービスオブジェクトが自身の処理を行う中で他のサービスオブジェクトのサービスを必要とする時、オブジェクト間通信が行われる。

通信機構の部品化を実現する上で、プリミティブな機能もオブジェクトの形にカプセル化した。以下で各オブジェクトの働きと他のオブジェクトとの関係を述べる。

4.1 サービスオブジェクト

開発者は各サービスをサービスオブジェクトにコーディングする。他のサービスオブジェクトへの要求はソースコード中に呼出し部を埋め込むことで行われる。通信に関わるコーディングは必要なく、これによりサービス部分を通信部分から分離する。

4.2 仮想オブジェクト

各サービスオブジェクト対応の通信部分を仮想オブジェクトとしてカプセル化し、通信の詳細をサービスオブジェ

*An Implementation of Communication Mechanism on DIABASE, Nobuhiro YOSHITAKE, Tsuyoshi MIYAMOTO and Satoru UEMICHI, NEC Corporation

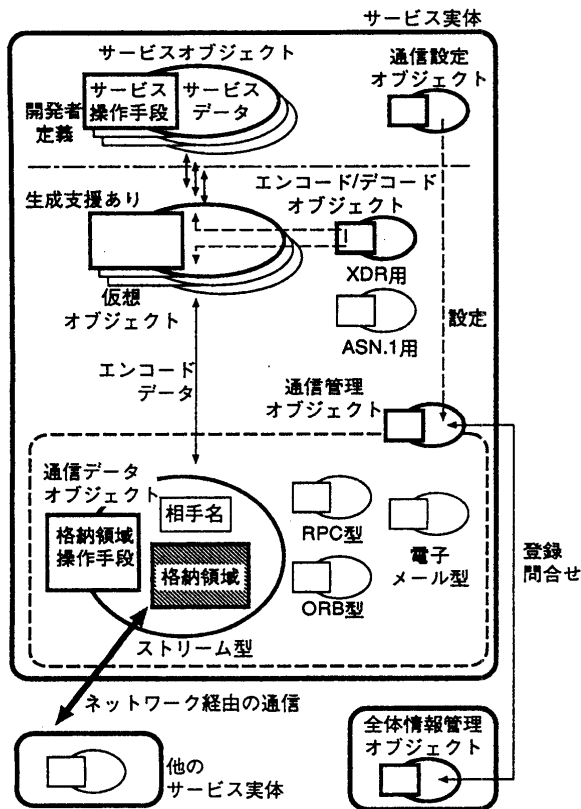


図 1: DIABASE の通信機構

クトへ見せない。これは遠隔手続き呼出し (RPC) のスタブとはほぼ同等の機能を提供する。異なるのは操作手段の管理でなく、データと操作手段をまとめたオブジェクト単位の管理を行う点である。

送受信するデータは仮想オブジェクト内でエンコード/デコードされる。その処理もオブジェクトとしてカプセル化され、オブジェクト選択が処理内容を決定する。仮想オブジェクトはビルド機構で生成支援される。

4.3 通信データオブジェクト

これは通信プロトコルごとに用意され、その選択が通信プロトコルを決定する。このことから通信プロトコル特有の処理は全て通信データオブジェクトの中にカプセル化される必要がある。

仮想オブジェクトから見えるのは通常、

- 送受信する相手サービスオブジェクトの識別名、
- 送受信したい (エンコードされた) 通信データを格納するための領域、
- その格納領域を管理する操作手段

である。通信データの格納操作手段が同時に相手サービスオブジェクトへのデータ転送を行い、データ取り出し手段が同時にデータ受信処理も行う。これにより、通信機構を単なるデータの入出力として扱え、通信プロトコルから独立した構造になる。

各通信プロトコルには特有のパラメータ設定が必要なが多いが、設定は通信管理オブジェクトを用いる。

4.4 通信管理オブジェクト

通信データオブジェクトを直接に監視する立場で、通信機構全般を把握するオブジェクトである。その役割は

- 通信 (開設 / 受信待ち / 送受信中 / 切断) の監視、
- サービスオブジェクトの動作 (サービス待ち / サービス実行中) の監視、
- 相手サービスオブジェクトの識別名と実アドレスのマッピング、
- 通信プロトコル依存の通信パラメータ設定と管理

である。通信パラメータ設定はデフォルト値を使わない限り開発者自ら何らかの形で行う必要がある。DIABASEの通信機構では、開発者が設定部分をコーディングする方法を用いるが、サービスオブジェクトとは別の特別なオブジェクト (通信設定オブジェクト) として実装する。これは、サービス部分と通信部分の分離の基本方針に従うためである。

4.5 全体情報管理オブジェクト

ネットワークで一意に管理したい情報を格納するオブジェクトである。ネットワークに一つ存在し、分散 AP の各サービス実体から必要に応じて、情報の登録 / 問合せが行われる。情報の種類として

- どのサービスがどのコンピュータで行われるか
- サービスオブジェクト間でどの通信プロトコルが使用可能か

等が格納される。

5 おわりに

DIABASE で分散 AP へ提供する通信機構を基本方針に従った形で検討、設計した。現在、通信プロトコルとしてストリーム型と RPC 型の二種類を用意し、実際の分散 AP 開発に利用して使い勝手、利用限界等を探っている段階である。

今後は、DIABASE のビルド機構の検討とともに通信機構の適用範囲を広げる予定である。

参考文献

- [1] 芳竹他, "分散アプリケーション開発環境の検討," 情報処大全第 44 回予稿集, p.5-167, 1992 年 3 月。