

分散オブジェクト環境向けクライアントシミュレータ*

5U-8

河村 謙治*、谷山 裕計*、花谷 直希*、柴山 武彦*、田中 吉廣**

*日立中部ソフトウェア(株)、**(株)日立製作所 ソフトウェア開発本部

1 はじめに

分散オブジェクト環境の普及により、ネットワークを意識せずに他マシン上のプログラムにアクセスする分散オブジェクト技術が注目されている。このような状況のもとで、分散オブジェクト環境のソフトウェアを開発する機会が増加してきた。同時に、分散オブジェクト開発のための支援環境も求められてきた。

本研究では、CORBA 規約に基づいた分散オブジェクト環境で動作する汎用的なクライアントシミュレータを開発した。本報告では、実現方法とシミュレータの機能について記述する。

2 目標

本シミュレータは分散オブジェクト開発の工数削減をねらって開発した。そのため、(1)汎用的であること、(2)操作が簡単であることを目標とした。汎用的であるとは任意のサーバに対して利用できることであり、操作が簡単であるとはサーバが変わる毎にコンパイル、リンクを必要としないことである。

3 実現方法

3.1 静的起動と動的起動

分散オブジェクト環境においてクライアントがサーバの関数を呼び出す方法は2つある。1つは静的起動である。これはC++の関数呼び出しとして記述するので簡単であるが、呼び出し関数が増えるとその都度コンパイル、リンクが必要という欠点がある。

もう1つは、関数呼び出しのリクエストをクライアントで組み立てて、動的起動インタフェースを利用して実行する動的起動である。静的起動と違って再コンパイルの必要はないが、サーバのイ

ンタフェース情報を与えておく必要がある。

本シミュレータの設定目標を実現するには、呼び出し関数が変わる毎に再コンパイルが必要な静的起動の方法は使えない。それ故、動的起動を選択することになるが、動的起動に必要なインタフェース情報は複雑で、操作性を損なうことなくそれを取得する方法が課題となった。

3.2 インタフェースリポジトリの利用

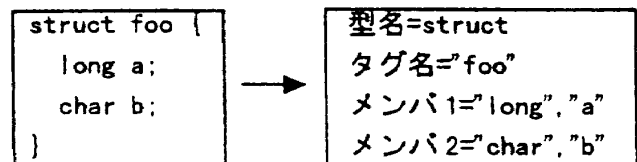
サーバのインタフェース情報を取得するために、インタフェースリポジトリを利用した。サーバのIDL(Interface Definition Language、インタフェース定義言語)ファイルをインタフェースリポジトリに入力し、その内容を解析することによって動的起動に必要なインタフェース情報を得ることができた。

3.3 リクエストの構築

インタフェース情報を得た後に必要なことは、関数呼び出しのリクエストを構築することである。これは主に引数を設定することである。

本シミュレータが汎用的であるためには、サーバのあらゆる情報を動的に管理せねばならない。例えば、構造体の引数があっても、これを構造体そのものとしては扱えない。ある型に特化した処理を記述した時点で汎用性が失われる。

これを解決するために、本シミュレータではCORBAの持つ型と同様の型を内部で定義した。例えば、構造体は次のように格納される。



この方法により、サーバの持つ情報を動的に扱うことが可能となり、汎用性を持たせることがで

* A Client Simulator for the Distributed Object Environment

Kenji Kawamura, Hirokazu Taniyama, Naoki Kajiya, Takehiko Shibayama, Yoshihiro Tanaka
Hitachi-TS; Software Development Center, Hitachi, Ltd.

5030 Totsuka-cho, Totsuka-ku, Yokohama, Kanagawa 244-0003, Japan

きた。しかしながら、独自の型表現を用いることにしたため、リクエストの構築に ORB(Object Request Broker)の API が使えず、API と同等の処理を記述しなければならなかった。上の例で、API を使える左の構造体と同じ処理結果を、右の独自形式で生成した。

以上の方法を用いることにより、再コンパイルすることなく、サーバの IDL ファイルのみで動作する汎用的なクライアントシミュレータを実現した。(図 1 参照)

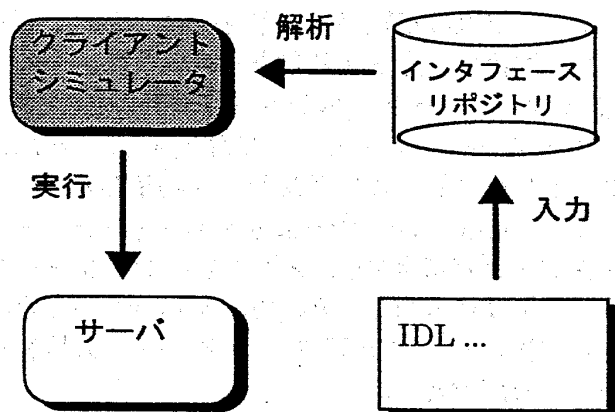


図 1 シミュレータの動作環境

4 シミュレータの機能

本シミュレータの機能を以下に示す。

(1) 単体テスト支援

サーバの関数の引数に値を対話的に設定してサーバを呼び出す。

(2) 性能テスト支援

関数の実行時間測定、結果のログファイル出力、繰り返し実行を行う。

(3) 再テスト支援

引数の設定値をファイル保存し、後に再利用できる。

(4) 負荷テスト支援

多クライアントを想定した並列実行を行う。

(5) プレイバック実行

実行結果を出力したログファイルを読み込み、同じ条件で再テストする。

(6) オブジェクトリファレンスの特定

サーバが生成したオブジェクトリファレンスの関数を呼び出すことができる。

(パラメタ表示部分の画面を図 2 に示す)

5 まとめ

分散オブジェクト環境向けのテスト支援ツールはまだ少ない。本シミュレータを使用することによって、クライアントのテストプログラムを作成する必要がなくなり、分散オブジェクト環境での分散オブジェクト開発が容易となる。今後、利用者の評価を頂き、機能改善や操作性向上を図っていく。

参考文献

- [1] 柴山、他：状態遷移表を用いた分散オブジェクト環境向け設計支援環境の構築、情報処理学会第 57 回全国大会論文集(1988)

型	名前	値
boolean		TRUE
struct phone {	phone_number_info	
string	name	TARO
string	number	045-123-4567
}		

図 2 パラメタ表示画面