

OSI RPC の実装に関する検討

3 P - 2

星加 修 内山 光一
(株) 東芝 情報処理・機器技術研究所

1. はじめに

ダウンサイジング、マルチベンダ化が進む中、高度な相互運用性を目指しオープンな標準化団体であるUI、OSFが分散処理環境の標準化を行なっている。その基盤技術としてRPC(Remote Procedure Call)を採用した。また一方で、ISOが標準化を行なっているOSIの実用化が盛んに行なわれている。ここでもFTAMやMHSから更に高度な分散処理が求められ始めた。この様な動きに応じてISOでは1990年よりOSI RPC^[1]の標準化を行なっている。そこで報告者らは今後、OSI環境での分散処理の基盤技術として期待されるOSI RPCの実装に関する検討を行なったので本稿で報告する。

2. OSI RPCについて

2.1 OSI RPCの特徴

表1において、RPCの重要な概念をサンマイクロシステムズのONC RPC^[2]と比較することによってOSI RPCの特徴を示す。

表1 OSI RPCとONC RPCの比較

	OSI RPC	ONC RPC
インターフェース	IDN	RPCL/rpcgen
バインディング	規定していない	ポートマップ
下位層	OSI	TCP/UDP
セキュリティ	規定していない	UNIX,DES
データ表現	ASN.1	XDR
セマンティクス	at most once	at most once
例外処理	エラー通知、キャンセル	エラー通知、タイムアウト

表1からわかるように、OSI RPCでは今後の検討項目になっているバインディングやセキュリティなどまだ規定されていない部分はあるが、同じような概念はあり機能的には大差無いと考えられる。

2.2 OSI RPCの通信モデル^[1]

OSI RPCの通信モデルを図1に示す。各構成要素について簡単に説明する。

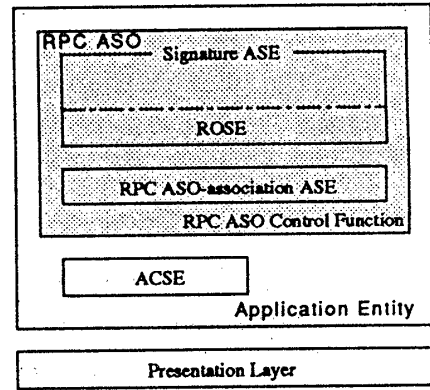


図1 OSI RPCの通信モデル

1) RPC ASO

OSI RPCのユーザにサービスを提供する。

2) Signature ASE

Signature ASEはRPCのプロトコルのうちアプリケーションに特有な部分である。この中には、要求/応答の相互動作に関するサービスを行うROSEを含む。Signature ASEとROSEのサービスは1対1に対応していて、Signature ASEのサービスをROSEのサービスにマッピングする。

3) RPC ASO-association ASE

ASOアソシエーションの確立と管理を行う。RPC ASO内に唯一存在する。

4) RPC ASO Control Function

RPC ASO内の構成要素の制御及び外部で発生したイベントの内部の構成要素へのルーティングを行う。

3. OSI RPCの実装

既存のRPCと同様に、RPCではアプリケーション開発者にリモートの手続きを意識させないことが重要であると考え、OSI RPCとしてドキュメントに規定されているもの以外に以下の2つの開発を行う。

- ・RPCコンパイラ

・バインディング機構

上記2つとOSI RPCとして規定されているプロトコルマシンを含む実行時にコールされるライブラリを開発したので以下に詳細を述べる。

1) RPCコンパイラ

表1に示したように、OSI RPCではIDN(Interface Definition Notation)と呼ぶサーバがどのようなサービスを提供できるかを記述するための言語を規定している。RPCコンパイラはIDNを入力とし図1のSignature ASEとクライアントとサーバで共有するファイルを自動生成する。(図2)

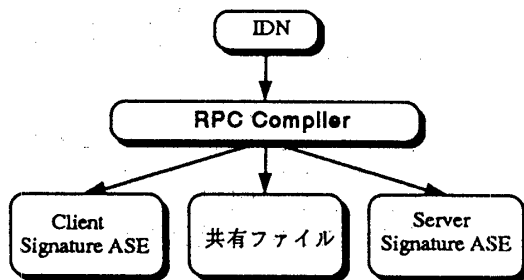


図2 RPCコンパイラ

なお、モデル上ROSEはSignature ASEに含まれているが、実装では別モジュールとして開発しSignature ASEからROSEをコールするようにした。(図3)

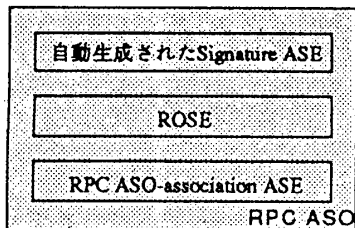


図3 RPC ASOの実装

RPCコンパイラが生成したクライアント側とサーバ側のSignature ASEには手続き呼び出しの引き数と結果のmarshallingとunmarshalling、ROSEへのマッピング、バインディング機構にアクセスするためのルーチンなどを含む。また共有ファイルはIDNで定義された値や型の宣言を含む。なお、Signature ASEと共有ファイルはC言語のソース・ファイルであり、OSI RPCではIDNからASN.1(Abstract Syntax Notation.1)へのマッピング規則は規定されているが、特定のプログラミング言語へのマッピング規則は規定していない。よって、IDNからC言語へのマッピング規則を独自に定めた。図4にマッピングの例を示す。図4は選択型と不定長の境界を持つ配列型の例である。なお、選択型のdiscrimと配列型のaという

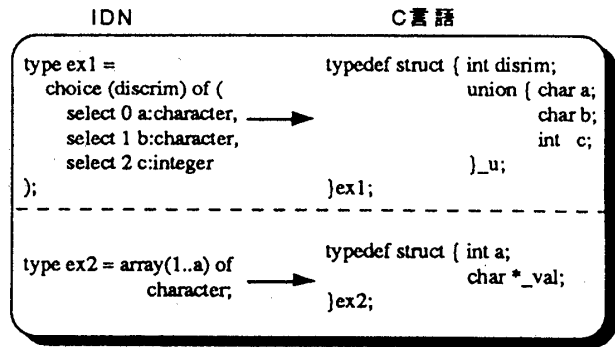


図4 IDNからC言語へのマッピングの例

2つの変数はその前に定義されているものとする。選択型はC言語の構造型に、選択型の要素は共用体にマッピングし、共用体は常に_uで参照される。可変長配列は、境界の値とその配列へのポインタを持つ構造体にマッピングする。

2) バインディング機構

分散処理において重要な概念である位置透過性を実現する。今回の開発ではクライアントはサーバの位置を意識する必要はないが、サーバ名を知っていなければならないというレベルの位置透過性を実現する。このバインディング機構は個々のサービスに与えられるオブジェクト識別子(インターフェース・タイプ識別子)とサーバ側のアドレスの管理を行なう。すべてのマシン上にデーモンとして存在する。

3) 実行時ライブラリ

図1のRPC ASO-association ASEのプロトコルマシンを含み、実行時にアプリケーションや生成されたSignature ASEからコールされる。

4. まとめ

OSI RPCを既存のRPCと比較することで仕様を明確にした。更に、OSI RPCのSignature ASEを自動生成するRPCコンパイラ、実行時ライブラリ、及びバインディング機構の実装を行なった。また、RPCコンパイラを開発する際に必要であるIDNからC言語へのマッピング規則を独自に定めた。今後は評価を行ないその結果を基に、より高度な分散処理の実現のための基盤技術となる様に、改良を続けていく予定である。

<参考文献>

[1]ISO/IEC 2ndCD 11578 "Remote Procedure Call"
 [2]Sun Microsystems,Inc. "RPC:Remote Procedure Call,Protocol Specification",RFC1050,1988