

分散計算機管理システムの設計

2C-5

宮内直人, 相澤 雅彦, 中川路 哲男, 勝山 光太郎

三菱電機 (株) 情報システム研究所

1 まえがき

分散コンピューティングの研究が進展し、それらを実現するソフトウェアが発表されている。分散コンピューティングを構成する要素として、ネットワーク管理とシステム管理の技術が重要である。我々は、分散コンピューティングシステムを管理するためのソフトウェアを設計した。本システムでは、マネージャ間の通信にはCMIPを採用し、マネージャと末端エージェント間の通信については、監視するためにSNMPを、制御するためにRPCを採用している。

本発表では、分散計算機システムを管理するソフトウェアの基本設計について報告する。

2 システム・アーキテクチャ

我々は、前回の発表[1]で、OSI管理による分散計算機システムの基本的な管理方式を示した(図1)。

このモデルでは、マネージャがOSI管理によってインテリジェント・エージェント(IA)を管理し、IAがRPCまたはSNMPを使ってシステム・エージェントとSNMPエージェントを管理する。

3 通信プロトコル

IAとエージェント間の通信プロトコルとして、RPCとSNMPの組合せから考えられる4つの案を、次の3つの観点から比較検討した。比較結果を表1に示す。

● 実現性評価

実装の難易度を評価するため、既存のプロトコルと被管理対象の変更の程度を評価した。

プロトコルや被管理対象の変更が必要ない場合は○。プロトコルや被管理対象の変更は必要ないが、複雑な実装方法をとるものは△。プロトコルや被管理対象の変更を伴う場合は×。

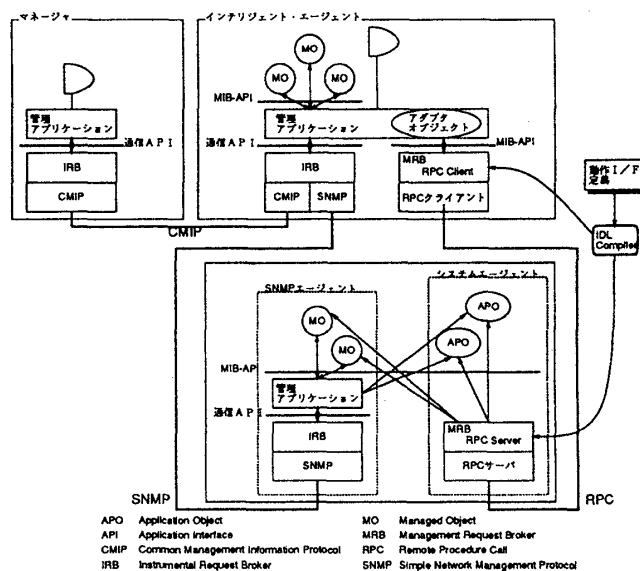


図1: 分散システム管理のアーキテクチャ

● メモリ性能(重さ、一貫性)評価

プログラムの大きさを評価する。プログラムの大きさは、実装するプロトコルの数に比例するものとして評価した。

実装するプロトコルが1種類ならば○、2種類ならば△、3種類以上ならば×。

● 拡張性評価

管理の拡張性と相互接続性を評価する。現在、ネットワーク機器や計算機の多くはSNMPを実装しているが、一方で、RPCをサポートしているネットワーク機器はほとんどなく、RPCで管理をしている例は少ないので、SNMPとの接続性を評価した。

SNMPマネージャからアクセスできる、またはSNMPエージェントにアクセスできれば○。一部分アクセスできれば△、全くアクセスできなければ×。

案1: プロトコルが2種類必要であるため実装が重くなるが、技術的には問題無く、既存のネットワーク機器も収容できる。

表 1: 通信プロトコルの比較

案	制御		監視		技術	重さ	拡張
	ネット	計算機	ネット	計算機			
1	RPC		SNMP		○	△	○
2	SNMP	RPC	SNMP	RPC	○	△	△
3	RPC				△	○	×
4	SNMP				×	○	○

案 2: プロトコルが 2 種類必要であるため実装が重くなり、計算機の管理については相互接続性が無いが、技術的には問題無く、既存のネットワーク機器も収容できる。

案 3: プロトコルが 1 種類に統一できるが、RPC によって非同期通知を行なう方法、及び既存のネットワーク機器の収容方法が問題である。

案 4: SNMP によって制御を行なう方法が問題であるが、プロトコルが 1 種類に統一でき、拡張性も高い。

上記のように比較した結果、案 1 を採用し、マネージャ/エージェント間の通信を表 2 のように決定した。

表 2: 通信プロトコル

	マネージャ間	マネージャ/エージェント間
制御	m.action	RPC
通知	m.eventReport	SNMP (Trap)
監視	m.get	SNMP (Get, GetNext)
設定	m.set	SNMP (Set)

4 管理情報

次の 3 段階に分けて管理情報の設計を行なった。

1. IA / 各エージェント間の管理情報の設計

SNMP エージェントの管理情報は、ネットワーク機器と計算機 H/W と S/W である。我々は、IAB が規定している MIB-II とホスト資源 MIB を使用し、さらに IAB に規定されていない管理情報 (OS, DBMS, TP モニタなど) については、IAB-SMI のフォーマットに従ってグループと属性、及び通知を定義した。

システムエージェントの管理情報は、アプリケーション・オブジェクト (APO) が用意する管理インタフェース (関数とコマンド) である。我々はこれらのイン

タフェースを IDL で記述し、IDL コンパイラによって、RPC サーバ・スタブとクライアント・スタブを生成した。

2. 上位/下位マネージャ間の管理情報の設計
OSI-SMI (GDMO) に従って、管理情報を設計した。
3. 上位マネージャ/下位マネージャの管理情報の変換テーブル設計
上記 1 と 2 で定義した管理情報の変換テーブルを設計した。変換テーブルは、MELMANAGER[2] で開発したものを拡張して使用した。

- 属性変換テーブル: OSI-SMI と IAB-SMI を変換する。
- 動作の変換テーブル: CMIP アクションを APO が提供する管理インタフェースに変換する。属性変換テーブルと共通の構造を持つ。
- 名前変換テーブルとアドレス変換テーブル: OSI, SNMP, RPC の名前とアドレスを変換する。
- 通知変換テーブル: SNMP Trap を CMIP のイベントに変換する。

5 アプリケーション・インタフェース

上記のように検討した結果、下位マネージャとエージェントの通信に 2 種類の管理プロトコルを採用したが、将来的にいずれか一方のプロトコルでも管理することを考慮して、MO と APO にアクセスするための API を共通化した (MIB-API)。

また、SNMP 及び CMIP を使用するための通信 API として、X/Open が規定している XMP-API を採用した。

6 むすび

分散計算機環境を管理するための管理アーキテクチャを報告した。現在、アーキテクチャに基づいた仕様設計を終え、IA とシステムエージェント、SNMP エージェントの製作を行なっている。今後は、これらの試験と評価を行なうと共に、IA における CMIP/SNMP/RPC 変換処理部を設計する予定である。

参考文献

- [1] 宮内他: 情報処理学会第 48 回全国大会 "OSI 管理による分散計算機システムの管理方式" (1994.3)
- [2] 三浦他: 情報処理学会第 46 回全国大会 "統合ネットワーク管理システム MELMANAGER - OSI 管理による LAN ドメインの管理" (1993.3)