

IU-4

サービスベース管理システムの構成

深沢 友雄 荻野 正 田中 英彦 元岡 達

(東京大学工学部)

1. はじめに

我々は、独立性の高い計算機同士を接続して、それぞれに蓄積されているサービス(プログラム、データ等)を提供する時に、ユーザがその分散性に煩わされることなくサービスを利用できるシステムを「サービスベースシステム(SBS)」と名付け、その構成方法について検討・実装を試みてきた。

今回は、SBSを構成する時に網全体の管理や、サービスの実行を行なう「サービスベース管理システム(SBMS)」の構成について述べる。

2. SBMSの構成(図1)

SBMSは、各計算機内でサービスの管理・実行を行なう部分と、計算機間にわたるサービスを処理する時に管理・実行を行なう部分とにわけて構成する。前者を「局所管理機能(LMF)」、後者を「広域管理機能(GMF)」と呼ぶ。

2.1 LMFの構成

LMFは、各計算機で独立にサービスの管理を行ない、既存の計算機のOSと同様の動作を行なう。LMFの構成は、計算機毎に異なっていてよい。サービスの実体は、LMFの上で作成・管理され、LMFのみによっても、実行される。

2.2 GMFの構成

GMFは、網中の計算機間にわたるサービスを管

理・実行する上で必要な役割を果たす、SBS特有の機能である。GMFは、「サービスの定義部(SDD)」と「サービスの処理系(SPS)」からなる。サービスの定義部(SDD)は、

- ①サービス名と、その意味を記述した部分(ビュー)
- ②サービス名と、その定義を記述した部分(マップ)にわけられる。①は、人間が理解する為のものであり、②は、処理系が解釈する為のものである。

ビューは、3層に分けて記述する。3層の各ビューに対応して、マップを作成する。マップ上には、ビュー上の各サービス毎に、そのサービスを実現する時にどのサービスをどの様に用いるかを記述した定義体を記述する。ビューとマップの記述は、「サービス定義言語」によって行なう。ビューとマップは、「関係」として表現され、関係データベースを用いて管理される。

サービスの処理系(SPS)は、SDDのマップの記述に基づいて、サービス要求を解釈し、自計算機が提供するサービスは自計算機で実行し、他計算機のいずれかが提供するサービスは、他計算機にサービス要求を行ないサービスの応答を待つ。各処理系が同様の動作を行なう事により、サービス要求が、サービスの存在する計算機に伝わる。サービスの実体が存在する計算機では、サービスの実行が行なわれ、処理が順次、要求元に返され、最終的にユーザが結果を得る事ができる。

3. SBMSの実現

本章では、各計算機上に、SBMSの構成要素であるLMFとGMFを実現する方法を示す。

3.1 LMFの実現方法

LMFを実現するには、次の2つの方法がある。

1. SBMS専用のLMFを各計算機に作成する
 2. 既存のOSを用いる
1. によって実現すれば、各計算機のLMFを全て統一する事ができ、GMFとのインタフェースも任意に設定できるので、GMFの実現が容易になる。但し、既存のOSに相当する機能を全て作成する必要があり、LMFの実装自体には、手間がかかる。又、各計算機の既存のOSと異なる管理機能をもうける事になり、既存のOS上で既に管理・蓄積されているサービスを利用する事が困難になる。

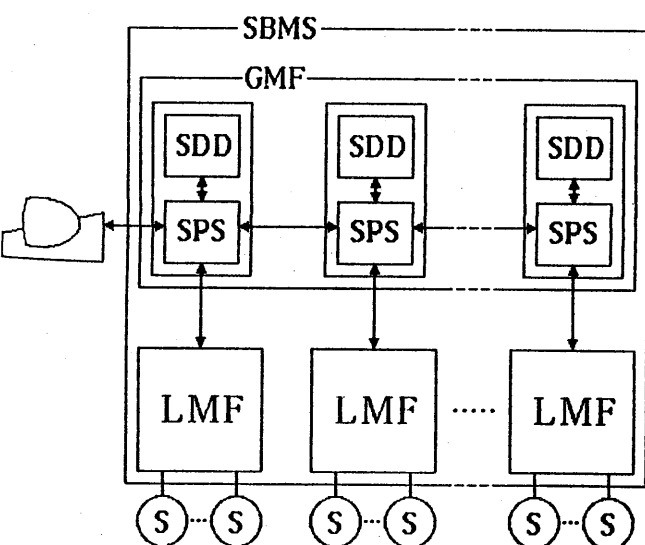


図1 SBMSの構成

一方、2. は、各計算機のOSの差異をGMFの処理系の部分で吸収する必要が生ずる。この為、或る計算機のOSが極めて特殊な場合、GMFを容易に構成できない可能性がある。しかし、各計算機の個々の管理機能は、各計算機の既存のOSが果たすので、LMFの実現は極めて容易である。

3.2 GMFの実現方法(図2)

GMFは、各計算機上の「処理系(SPS)」と「サービスの定義部(SDD)」からなる。SPSが網全体にわたるサービスの実行・管理を行なう時に、各計算機内だけで実行・管理できることについては、各計算機のLMFを用いる。従って、SPSの実現方法は、LMFの実現方法に依存する。LMFをSBMS専用で作成する場合は、LMFとSPSのインターフェースを自由に設定できる。この時、各計算機内だけでできる事は、LMFで行ない、その他の事はSPSで行なう様に設計する。LMFとして既存のOSを用いる場合、SPSは、サービス要求を解釈する部分(SLP)と、通信を行なう部分(CMP)から構成する。SLPは、OSが管理するプログラムの1つとして作成する。各計算機がSBMSのノードとして動き始めるということは、このプログラムが、プロセスとして起動される事に相当する。SPS同士は、互いに通信を行なう必要がある。SPSの行なう通信は、SLPのレベルの通信とCMPレベルの通信に分かれる。前者を上位レベルの通信、後者を下位の通信と呼ぶ。各計算機のOSは、SLPが下位レベルの通信機構を利用できる構造になっている必要がある。SDDは、「ビュー」と「マップ」からなる。これらは、各計算機のLMFが持つ二次記憶管理機能即ち、ファイル機能、あるいは、データベース機能を用いる。

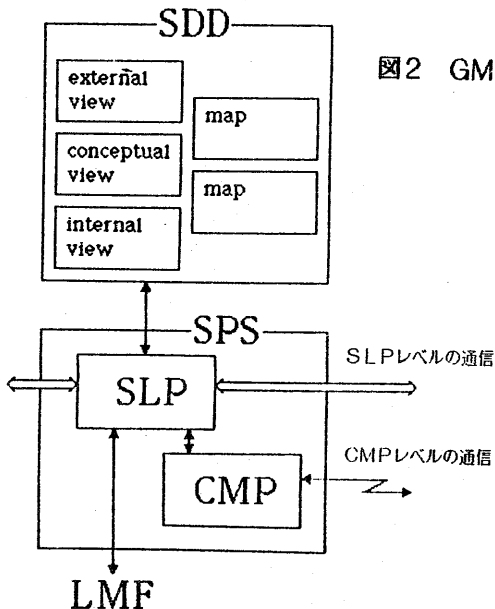


図2 GMFの実現

4. 実装システムのシステム構成(図3)

実験システムのハードウェア構成は、SBS内の各ノードとして、M280H、VAX-11/780、VAX-11/730を用いた。

M280Hのソフトウェア構成は、OSとしてVOS3を用いる。VOS3のTSSの提供する機能をLMFとする。データベースに関する機能は、C言語によって記述し、実行可能モジュールの形で動く。データベースの実体は、VOS3上の順編成ファイルによって実現する。GMFのうち、SLPは、サービス定義・要求言語を、関数型言語とした場合はUTILISP、論理型言語とした場合はC-Prologの処理系を用いる。又、LMFとのインターフェースの為に、SLPの一部はC言語で記述した。

2台のVAX-11のソフトウェア構成は、OSとしてUNIXを用いる。UNIXの提供する機能をLMFとする。LMFのうち、データベースに関する機能は、C言語によって記述し、UNIX上の実行可能モジュールの形で動く。データベースの実体は、UNIX上のファイルによって実現する。

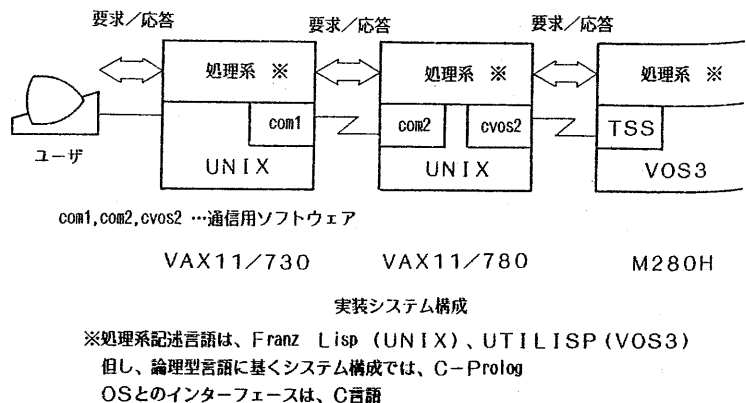
CMPは、VAX-11/730とVAX-11/780間は、730側は本研究室で開発したcomというプログラムを、780側はUNIXの提供する端末用のソフトウェアを用いる。VAX-11/780とM280H間は、780側はcvos2というプログラムを、M側は日立製のT560/20ビデオデータターミナル用のソフトウェアを用いる。

5. おわりに

本稿では、SBMSの構成とその実現方法を明らかにし、実験システムでの構成例を示した。今後は、ユーティリティの充実と応用サービスの拡充を行ない、SBSの有効性の確認を行なっていく予定である。

6. 参考文献

[1] 深沢、田中、元岡、「サービスベースシステム」、情報学会、分散処理学会研究会、1984.10



※処理系記述言語は、Franz Lisp (UNIX)、UTILISP (VOS3) 但し、論理型言語に基づくシステム構成では、C-Prolog OSとのインターフェースは、C言語

図3 実験システムの構成