

OSI 管理マネージャへのシステム管理機能の実装法

本村 公太[†] 平木 久美子[†] 高野 誠^{††}

ネットワークの大規模化・複雑化にともない、ネットワーク管理の重要性が増している。ISO ではネットワークに接続された開放型システムの管理 (OSI 管理) のために様々な標準の規定を進めている。その中でシステム管理機能 (SMF) は、様々な管理機能のために必要となるサービスとそれに関連する管理情報を定義している。本論文では、OSI 管理のマネージャ (ネットワーク管理装置) において、SMF のサービスを利用する管理アプリケーションの開発を容易にする SMF 実装法を提案する。まず、管理アプリケーションと一体化して SMF を実装するのではなく、共通的で基本的な管理サービスを提供するミドルウェアとして SMF を実装していくことを提案する。次に、そのミドルウェアを利用する上位の管理アプリケーションでの処理を容易にするために提供する付加機能と、上位管理アプリケーションの開発者が使いやすい API の規定法を提案する。さらに、拡張性の高いプログラム構成と、実装上重要な処理方法をいくつか提案する。最後に本論文の提案内容に基づいてワークステーション上にプログラムを試作した結果に基づき、有効性と今後の課題を示す。

An Implementation Method of Systems Management Functions for an OSI-based Manager

KOTA MOTOMURA,[†] KUMIKO HIRAKI[†] and MAKOTO TAKANO^{††}

Network management becomes increasingly important as networks are globalized and become more complicated. ISO has been specifying various standards for management of open systems. Systems management functions (SMFs) define the services necessary for achieving various management functions and the management information related to them. In this paper we propose an implementation method of SMFs in an OSI-based manager (network management station) for easy development of management applications which use the services of SMFs. First of all, we propose a SMFs-based middleware which provides basic and common management services to upper level management applications and make clear issues in the implementation of SMFs. There are two main issues: to provide additional functions to enable the easy processing in those applications and to provide user friendly APIs for developers of them. Next we show how to define the APIs and propose some additional functions. We also propose a flexible program structure for the implementation of SMFs and some technical points in the implementation. Finally, prototype evaluations and issues for further study are described.

1. はじめに

ネットワークの大規模化・複雑化にともない、ネットワーク管理の重要性が増している。ネットワークに接続された開放型システムの管理のための国際標準として、OSI (Open Systems Interconnection) 管理に関する様々な標準化が進められている。これらの標準には管理情報転送のための共通プロトコルである CMIP (Common Management Information Protocol)¹⁾ とそのためのサービス定義である CMIS

(Common Management Information Service)²⁾、管理情報の定義方法を規定する GDMO (Guidelines for Definition of Managed Objects)³⁾、様々な管理機能のために必要なサービスとそれらに関連する管理情報を定義するシステム管理機能 (SMF: Systems Management Function)⁴⁾ などがある。SMF には状態管理、警報報告、イベント報告管理、ログ制御など現時点で約 20 種類あり、国際標準の段階に達したものからドラフト段階のものまでである。

OSI 管理はキャリア系のシステムを中心に実装が進みつつあり、それらのシステムでは必要に応じて SMF についても限定的に実装している⁵⁾。また OSI 管理に関する汎用的な実装法としては、GDMO に基づく管理情報を CMIP で受け取って GUI (Graphical User

[†] NTT 通信網研究所
NTT Telecommunication Networks Laboratories

^{††} NTT サービス生産本部
NTT Service Engineering Headquarters

Interface) 画面に表示するという点に関して、管理アプリケーション (以下、管理 AP と略す) の作成を容易にするために様々なツールを持つプラットフォームも提案されている⁶⁾。また SMF の実装法としては、GDMO で定義された SMF の管理オブジェクト (MO: Managed Object) の属性情報等の符号化と復号のためのスタブを自動生成するツールも提案されている⁷⁾。OSI 管理で扱われる MO はすべて GDMO で定義されるので、GDMO の面での扱いを容易にすることは基本的かつ重要なことである。しかし、SMF のサービスを利用して高度な管理サービスを提供する様々な管理 AP の開発を考えた場合、それだけでは不十分で、管理 AP での処理を容易にするための様々な付加的なサービス (たとえば、過去のイベント情報やトポロジ情報を管理するデータベース機能など) も提供する必要がある。また SMF では CMIS のサービスをそのまま SMF のユーザに提供するパスルーサービスを定義し、それを利用することが多いが、汎用的であるがゆえに管理 AP の開発者が使いこなすのは難しい。このため、管理 AP の開発者にとって意味的に分かりやすい、使いやすい API (Application Programming Interface) を提供することも必要である。

本論文では、OSI 管理のマネージャ (ネットワーク管理装置) において、SMF のサービスを利用する管理 AP の開発を容易にする SMF の実装法を提案する。2 章では、本論文の理解を助けるために SMF の規定の概要を述べる。3 章では、共通的で基本的な管理サービスを提供するミドルウェア (基本管理 AP と呼ぶ) として SMF を実装していくことを提案し、そのための課題と対処方針を明らかにする。次にその対処方針に従い、基本管理 AP を利用する上位の管理 AP での処理を容易にするために提供する付加機能 (SMF 規定外の機能) と、上位管理 AP の開発者にとって使いやすい API の規定法を提案する。4 章では、拡張性の高いプログラム構成と、実装上重要な処理方法を提案する。5 章では、提案内容に基づいてワークステーション上にプログラムを試作した結果に基づき、評価と考察を述べる。最後に提案内容と今後の課題についてまとめる。

2. システム管理機能 (SMF) の概要

本章では、3 章以下に示す提案の理解を助けるため、SMF の位置付けと規定例について述べる。OSI 管理の枠組みや CMIP/CMIS の詳細については、解説書 (たとえば文献 8)) 等を参照されたい。

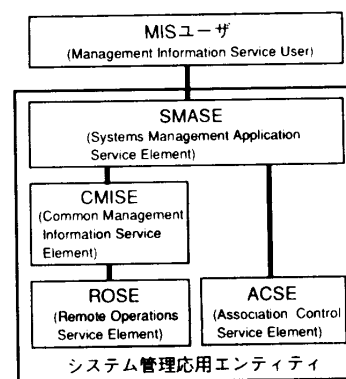


図1 システム管理に関する OSI 応用層の構造
Fig.1 Structure of OSI application layer related to systems management.

2.1 SMF の位置付け

OSI 管理における SMF の位置付けを明確にするために、システム管理に関する応用層の構造を図 1 に示す。システム管理応用エンティティは、ACSE, ROSE, CMISE, SMASE の 4 つのサービス要素から構成される。SMF は SMASE の中で使用される。SMASE は、ACSE を使用して相手エンティティとアソシエーションを確立し、CMISE を利用して管理操作/通知を行い、MIS ユーザに管理サービスを提供する。CMISE が提供するサービスが CMIS で、使用するプロトコルが CMIP である。CMIS では、MO*の生成と削除、MO の属性値の取得と設定、属性値取得の取消、動作指示、イベント報告の 7 種類のサービスを定義している。SMF で規定されるサービスは CMIS にマッピングされ、SMF の管理情報は CMIP で運ばれる。SMF 利用時の SMASE の動作は、マネージャ (管理側) の役割とエージェント (被管理側) の役割に分けて記述されている。

2.2 SMF の規定

現在標準化が進みつつある SMF の一覧を表 1 に示す。ここでは、オブジェクト管理機能とイベント報告管理機能を例にとり、その規定の概要を述べる。

(1) オブジェクト管理機能

本機能は、SMF の中で最も基本的な機能を規定しており、MO の生成と削除、MO の属性値の取得と設定、属性値の変更にもなう通知等を定義する。具体的には、属性値取得の取消以外の 6 種類の CMIS のサービスをそのまま MIS ユーザに提供するパスルーサー

* MO クラスは抽象的な定義であり、実際に生成されるのは MO インスタンスである。本論文では、MO とのみ記した場合は MO インスタンスを指すものとする。

表1 SMFの標準化状況(1995.5現在)
Table 1 Status of standards for SMFs.

番号	タイトル
IS 10164-1	オブジェクト管理機能
IS 10164-2	状態管理機能
IS 10164-3	関係表現のための属性
IS 10164-4	警報報告機能
IS 10164-5	イベント報告管理機能
IS 10164-6	ログ制御機能
IS 10164-7	セキュリティ警報報告機能
IS 10164-8	セキュリティ監査証跡機能
IS 10164-9	アクセス制御用管理オブジェクトおよび属性
DIS 10164-10	会計利用機能
IS 10164-11	計測オブジェクトおよび属性
IS 10164-12	試験管理機能
IS 10164-13	集計機能
DIS 10164-14	信頼性と診断試験カテゴリ
IS 10164-15	スケジューリング機能
DIS 10164-16	管理知識管理機能
DIS 10164-17	切替機能
DIS 10164-18	ソフトウェア管理機能
CD 10164-19	管理領域管理ポリシー管理機能
CD 10164-20	時間管理機能

IS: International Standard,

DIS: Draft International Standard,

CD: Committee Draft

ビスと、特定の通知としてMO生成報告、MO削除報告、属性値変更報告の3種類のサービスを定義している。パススルーサービスは、他のSMFで必要となる特定のMOの生成等のためにも頻繁に使用される。

一方、管理情報としては、上記の特定の通知のためのイベントタイプとイベント情報、それらをログとして記録するためのMOクラスなどを定義している。

(2) イベント報告管理機能

本機能はエージェントから受け取るイベント報告をマネージャが制御するための機能を定義する。具体的には、イベント転送選別器(EFD: Event Forwarding Discriminator)というMOクラスと、関連する動作を定義する。マネージャがEFDの生成をエージェントに対して要求することにより、イベント報告が開始される。どのようなイベント報告を受け取るかは、ある属性値がある値に等しいというような条件を論理演算子(AND, OR, NOT)でつなぐ形で選別器コンストラクト属性において指定する。その他の属性としては、選別器の識別子、運用状態、イベント報告の宛先、開始と終了の時刻、様々なスケジューラ、バックアップ宛先リスト等がある(個々の属性の内部構造や指定可能な属性の組合せの制限もあるが、説明は省略する)。

特定のサービスの定義はなく、すべてパススルー

サービスを使用する。すなわち、イベント報告の開始要求は前述のとおりEFDの生成要求となり、終了はEFDの削除となる。転送条件の読み出しには属性値の取得要求、変更には設定要求を使用する。また、イベント報告の一時停止や再開は、EFDの運用状態属性の値を属性値設定要求により書き換えることにより行う。

3. システム管理機能の実装法

本章では、筆者らが提案するSMFの実装方針について述べ、実装のための課題と対処方針を明らかにする。次に、その対処方針に従い、上位の管理APでの処理を容易にするために提供する付加機能と、上位管理APの開発者が使いやすいAPIの規定法を提案する。

3.1 SMFの実装方針および課題と対処方針

(1) 実装方針

SMFの実装方針は以下の2つに大別できると考える。方針1が従来の実装方針に対応し、方針2が筆者らの提案する実装方針である。

方針1: SMFを管理APと一体化して実装することを前提にして、SMFで扱う管理情報の符号化と復号の部分を支援する。

方針2: 様々な管理APで使う共通的で基本的な管理サービスを提供するミドルウェア(基本管理AP)としてSMFを実装する。

まず方針1の考え方について述べる。SMFのサービスはCMISで受け渡しする情報とその意味を特定しているだけなので、図1でのSMASEとMISユーザを一体化してCMIS上の管理APとして実装することができる。情報の解釈はもともと管理APの仕事となるため、特定の目的の管理APの場合は一体化したほうが作成しやすく、その際にSMFで扱う管理情報の符号化と復号を容易にするための手段を提供しておくことが有効になる。符号化と復号のためのスタブを自動生成する方法⁷⁾やGDMO用のツールを用いて管理プラットフォーム上にSMFを実装する方法⁶⁾は、この考え方をとっているものと見なすことができる。

次に方針2の考え方について述べる。大規模化・複雑化したネットワークの管理を容易にするために、知識処理等を利用した高度な管理APが作成されるようになってきている⁹⁾。管理APの種類と規模は、管理の高度化にともなって増加する一方である。もしそれぞれがSMF等の基本機能を個別に実装すると、重複する部分が無駄になり、規模もより大きくなる。そのため、共通的で基本的な管理サービスを提供するミドルウェア

アを SMF ベースに実装しておき、その上に必要に応じて高度な管理 AP を開発していくのが効率的であり、拡張性も高くなる。このような考え方から、方針 2 の実装方針を提案する。以下では、基本管理 AP が提供するサービスを基本管理サービス、基本管理 AP を利用する管理 AP をユーザアプリケーション（略してユーザ AP）と呼ぶこととする。

(2) 課題と対処方針

提案した実装方針を採用する場合、基本管理 AP として以下の 2 つの主要課題がある。

課題 1: SMF で規定されている以外で付加すべき機能の選定。

課題 2: ユーザ AP 開発者が使いやすい API の規定。
まず課題 1 の解決の必要性と対処方針について述べる。様々な種類のユーザ AP が独立に開発されることを想定すると、それらの独立性を維持するために、個々のユーザ AP を管理し、個別の動作要件を反映した動作を可能としておく必要がある。しかし、SMF は MIS ユーザを 1 つとしてモデル化しているため、これに対応する SMF の規定はない。また、ユーザ AP には、障害の原因分析のために障害発生時のみ起動されるものなど、非常駐のものも考えられる。そのようなユーザ AP にとって、起動されてからネットワークのトポロジ情報や過去のイベント情報を個々のエージェントから収集するのは困難であるし、時間もかかりすぎる。このような場合、基本管理 AP がそれらの情報を蓄積しておき、ユーザ AP に対して様々な検索サービスを提供するのが効果的である。しかし、SMF はマネージャ・エージェント間でのやりとりのみをモデル化しているため、マネージャで行う収集済の情報の蓄積・検索等はスコープ外である。以上のように、様々なユーザ AP を想定した場合に SMF だけでは不足すると思われる機能を選定し、それらを基本管理 AP が付加機能として提供していくことを提案する。提供すべき付加機能の詳細は 3.2 節で述べる。

次に課題 2 の解決の必要性と対処方針について述べる。基本管理サービスをユーザ AP に提供するためのインタフェースは、API として規定することになる。様々なユーザ AP の開発者がこの API を使用するため、開発者にとって使いやすい API とする必要がある。API の規定の仕方は次の 2 つの方針に大別できる。

方針 1: 汎用的な少数の API を規定する。

方針 2: 特定の用途ごとの API を規定する。

SMF では CMIS をそのまま提供するパスルーサービスを使用することが多く (2.2 節参照)、SMF の規定に単純に従うと方針 1 を採用することになる。

この場合 API の数が少なく済むので効率的ではあるが、1 つの API で指定可能なパラメータの種類は非常に多くなるため、開発者が CMIS や SMF に精通して API を使い分ける必要がある。しかし、すべての開発者にそれを要求するのは難しい。そのため、方針 2 を採用して API の意味を分かりやすくし、指定すべきパラメータ数も減らして提供することを提案する。方針 2 に従った API の規定法の詳細は 3.3 節で述べる。

この他にもユーザ AP の GUI 画面作成や GDMO 処理等を容易にするためのツール類を提供することも重要であるが、市販のツールが利用可能であったり、文献 6) 等で検討されているため、本論文の対象外とする。

3.2 提供すべき付加機能

前節の考え方を実現するために基本管理 AP が提供すべき主な付加機能を表 2 に示す。これらの機能は、従来の管理システムの機能をもとに様々なユーザ AP を想定し、ユーザ AP での処理やその開発を容易にできるものを選定した。

3.3 基本管理サービス API の規定法

ユーザ AP の開発者にとって使いやすい、特定の用途ごとの API の規定の仕方を以下に示す。

(1) 意味を表す API とする。

たとえば、SMF では EFD という MO の生成がイベント報告の受け取り開始となり (2.2 節参照)、ログという MO の生成がログの収集開始となる。これらの意味を表現するため、共通の MO 生成要求 API ではなく、イベント転送開始要求 API、ログ収集開始要求 API というように分けて規定し、それぞれに必要なパラメータも明確にして提供する。

(2) 識別名の代わりに整数の ID も使用可能とする。

SMF では操作対象の MO は識別名[☆]で指定するが、それをそのままプログラム間で受け渡しすると複雑になる。そのため、識別名は基本管理 AP で内部的に管理し、識別名に対応する一意の整数の ID を割り当て、ユーザ AP が ID も使用できるようにする。

(3) 複数 MO の指定にリスト形式も使用可能とする。

属性情報の読み出し等では、CMIS のスコープとフィ

[☆] 識別名 (FDN: Full Distinguished Name) は、MO の包含関係を表す階層構造の包含木に基づき、ルートから目的の MO までの枝の名前 (属性とその値) をたどって付与される。その結果、通常は長いデータ列となる。

表2 基本管理アプリケーションの主な付加機能
Table2 Major additional function for the basic management application.

付加機能名	機能概要
ユーザ管理機能	ユーザ AP の登録, 利用開始と終了, 個別の動作条件の設定, 他のユーザ AP に関する情報の検索等のサービスを提供する.
メッセージ管理機能	ユーザ AP 間でのメッセージの送受信を可能とするサービスを提供する.
ローカルフィルタ管理機能	各ユーザ AP が必要なイベント報告だけを受け取れるようにするため, ローカルなイベントフィルタサービスを提供する.
ローカルイベントログ管理機能	ユーザ AP が問題解析等のために過去のイベント報告を調査できるようにするため, 受け取ったイベント報告をログとして管理しておき, 各種検索サービスを提供する.
ネットワークトポロジ管理機能	ネットワークに接続された被管理システムやリンクに関する情報をデータベースに保持し, 状態変更のイベント報告等を利用して情報を更新することによって, 最新のトポロジ情報を提供する. 隣接したノードの検索サービス等も提供する.
定期属性値読み出し管理機能	ユーザ AP の指定した MO の属性値を定期的に読み出してログとして管理し, 各種検索サービス, 属性値があらかじめ指定された条件を満たした場合には警報のイベント報告として通知するサービスを提供する (主に低機能のエージェントの監視用).
自動管理オペレーション管理機能	指定されたイベント報告の受信を契機として, あらかじめ登録されたオペレーション (所定の様式で記述されたバッチファイル) を実行するサービスを提供する.
オブジェクト ID 管理機能	MO の識別名とそれに対してローカルに付与する整数の ID の対応を管理し, 相互変換のサービスを提供する.
トレース管理機能	ユーザ AP が発行した API, あるいは基本管理 AP が下位の通信機能等に対して発行した API のトレース情報をログとして管理し, ウィンドウに表示するサービスを提供する (主に管理システム全体のデバッグ用).

ルタ^{*}を使用できる. MO の識別名が不明のときには便利であるが, あらかじめ分かっている複数の MO を同時に指定するときには使いにくい. そのため MO の識別名のリスト形式および前述の整数 ID のリスト形式も使用可能とする. さらに, 異なるエージェントの MO でも指定可能とし, 基本管理 AP 内で展開して複数エージェントへの一括アクセスを可能にする.

(4) EFD の指定等を簡素化してパラメータ数を減らす.

EFD の生成に際しては様々な属性を指定する必要があるが, ユーザ AP によっては特定のタイプ (たとえば警報) のイベント報告だけを受け取る指定ができれば十分なものも多い. 一方, 数多くのユーザ AP がイベント報告開始要求を行う度にエージェントに対して細かな条件を付けた EFD の生成を要求すると, エージェント側の負荷もかなり大きくなる. これらの点を考慮し, 受け取りたいイベント報告のタイプなど簡単な条件設定だけが可能なローカルなイベント報告管理

のための API も規定し, こちらの使用を推奨する.

なお, 本節では分かりやすい API や付加的な API の提供を提案しているのであり, SMF の機能が API で制約を受けることはない. また, パススルーサービスに対応した API を提供することにより, 未サポートの SMF で定義された MO へユーザ AP がアクセスすることも可能となる. API の規定例については, 5.1 節で示す.

4. プログラム構成と実装上重要な処理方法

本章では, 拡張性の高いプログラム構成とともに, 実装上重要な処理方法を提案する.

4.1 プログラム構成

拡張性や実装の容易さを考慮して筆者らが提案する基本管理 AP の構成を図 2 に示す. 付加機能は, ユーザ AP ごとの要求に基づく処理を行うユーザ対応付加機能群と, ユーザ AP と同様に基本管理サービスを利用する形の内部ユーザ付加機能群と, 他の機能やインタフェース部から共通的に利用される共通付加機能群に分けて実装する. この構成によって, 各付加機能の位置付けやインタフェースを明確化し, 機能追加の際の影響範囲を限定することができる. 特に内部ユーザ

^{*} MO を間接的に指定する手段である. たとえば, 包含木上の特定の MO を基点にしてその下位の MO 全体をスコープで指定し, その中から特定の条件を満たす MO だけをフィルタで選ぶことができる.

表3 2種類の管理情報体系を対応付けたトポロジ情報の一部
Table 3 Example of topology data for mapping of two types of ensembles.

トポロジ情報	回線構成管理の管理情報体系	LAN 障害管理の管理情報体系
MO クラス	opEquipment クラス	internetSystem クラス
識別子	equipmentID 属性値	sysObjectID 属性値
状態	administrativeState 属性値, operationalState 属性値	administrativeState 属性値, operationalState 属性値
種別	functionList 属性値	ifEntry クラスの数
接続リンク MO クラス	circuit クラス	ifEntry クラス
リンク識別子	circuitId 属性値	ifIndex 属性値
リンク状態	administrativeState 属性値, operationalState 属性値	ifAdminStatus 属性値, ifOperStatus 属性値
物理アドレス	a_TPIInstance のローカル ID	ifPhysAddress 属性値

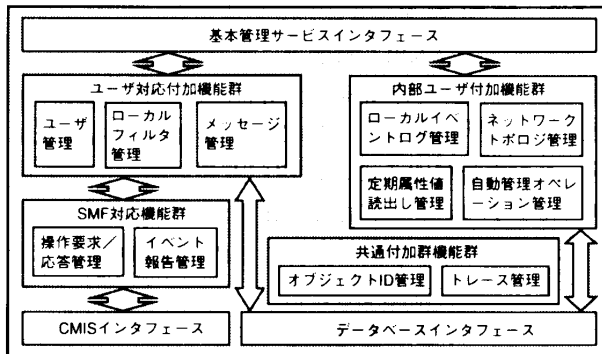


図2 基本管理アプリケーションの構造

Fig. 2 Structure of the basic management application.

付加機能として機能追加する場合、個々の付加機能は完全に独立して管理可能なので、拡張が容易になる。

4.2 ローカルフィルタの処理方法

基本管理 AP では、ローカルなイベント報告管理のための API を設け (3.3 節参照)、ローカルフィルタ管理機能 (表 2 参照) によって実現する。基本管理 AP は、被管理システムごとにデフォルトの選別器コンストラクトを持つこととし、立ち上げ時に各被管理システムのエージェントに対してそれに従った EFD の生成を要求する。これによってユーザ AP の起動前でもイベント報告の収集が可能となり、ユーザ AP は自分が起動される前のイベント情報でもローカルイベントログ管理機能 (表 2 参照) を利用して調査可能となる。

起動されたユーザ AP がそれぞれの要求に従ってローカルイベント報告開始要求 API を発行すると、基本管理 AP はその時点で設定している選別器コンストラクトと指定された選別器コンストラクトを比較する。もし前者で後者をカバーできない場合、基本管理 AP は、両者を OR 演算することによって新しい選別器コンストラクトを決定し、エージェントの EFD に設定する。エージェントからのイベント報告は、各ユーザ

AP の指定した選別器コンストラクトに合うものを選別して渡す。この実装によってエージェント側の負荷を軽減することができ、また、1つのエージェントに各ユーザ AP が独立に複数の EFD を生成した結果として同一のイベント報告が複数回送信されるのを防ぐことができる。

4.3 トポロジ情報の保持方法

ネットワークトポロジ管理機能 (表 2 参照) で保持する情報としては、被管理システムに関する情報と、それが接続しているリンクに関する情報を基本とする。具体的には、被管理システムやリンクの識別子、状態、種別、接続関係等を保持する。これらの情報は被管理システムの管理情報の体系によって異なるが、ユーザ AP がその違いを意識しなくても済むように統一したトポロジ情報として扱うことが望ましい。ここでは、Network Management Forum が定義した回線の構成管理用と LAN の障害管理用の 2 種類の管理情報体系^{10),11)}を対象とする。2種類の管理情報体系を対応付けたトポロジ情報の一部を表 3 に示す。回線のリンクと LAN のセグメントを等価なものに見なすことによって、2種類の体系を同一のテーブル形式で管理できる。

トポロジ情報は基本管理 AP の起動時に初期情報の収集を行い、その後は受信した MO 生成報告や状態変更報告から必要な情報を抽出して最新情報に更新する。

5. 評価と考察

4 章までの提案に基づき、CMIP をサポートした汎用ワークステーション (CPU は 50MHz の PA-RISC) 上に基本管理 AP を試作した。SMF としては、構成、障害、性能、セキュリティ管理に関して標準化の進展しているものから 11 種類 (10164-1~8, 12, 13,15) を実装した。本章では、試作から得た評価結果について考察する。まず、試作した基本管理 AP 上で障害管

理のアプリケーション¹²⁾等の試作を進めた経験をもとに、API 規定の妥当性及び付加機能の過不足について考察する。次に、試作プログラムの規模内訳を示し、今後の機能追加の際の規模見通しについて考察する。最後に、性能評価として実験環境でのユーザ AP レベルの応答時間を示し、整数 ID 付与のオーバーヘッドについて考察する。

5.1 基本管理サービス API

3.3 節の考え方に従って規定したイベント転送開始要求 API とローカルイベント転送開始要求 API を例として図 3 に示す。イベント転送開始要求 API は、必ずしも使いやすい API には見えないが SMF の規定を反映しているためである。パススルーサービスに対応した MO 生成要求 API を利用する場合、MO クラス等の指定が必要になる他、すべての属性は単なる属性リストとしてひとかたまりのデータとして扱うので、それよりは使いやすくなっている。一方、ローカルイベント転送開始要求 API は、パラメータが大幅に少なくなっており、使いやすくなっていることが分かる。

(1) イベント転送開始要求APIの関数形

Event Forward Start (User_ID, Req_ID, Sync_Mode, Local_ID, FDN, Superior_Local_ID, Superior_FDN, Reference_Local_ID, Reference_FDN, EFD_Attribute_List, Access_Control, Message)

関数パラメータ	文字列や構造体はポイントで指定する
User_ID	整数 / ユーザAPのID */
Req_ID	整数 / 要求ID */
Sync_Mode	整数 /* 要求の回答を待つ(同期)か非同期でもらうか */
Local_ID	整数 /* MOに与えるローカルID */
FDN	文字列 /* MOの識別名 */
Superior_Local_ID	整数 /* 上位MOのローカルID */
Superior_FDN	文字列 /* 上位MOの識別名 */
Reference_Local_ID	整数 /* 参照MOのローカルID */
Reference_FDN	文字列 /* 参照MOの識別名 */
EFD_Attribute_List	構造体 / EFDの属性リスト */
Access_Control	符号化文字列 /* アクセス制御のためのデータ */
Message	構造体 /* 要求の回答の受け取り(同期型の時) */

EFD_Attribute_Listの属性

Discriminator_ID	整数か文字列 / 選別器のID */
Discriminator_Construct	構造体 /* 選別器コンストラクト、フィルタ構造体 */
Operational_State (Read Only)	整数 /* 動作状態 (enabled, disabled) */
Administrative_State	整数 /* 運用状態 (locked, unlocked) */
Availability_Status (Read Only)	整数 /* 利用可能状態 (off duty, etc) */
Start Time	時刻構造体 /* イベント報告期間の開始時刻 */
Stop Time	時刻構造体 /* イベント報告期間の終了時刻 */
Interval Of Day	構造体 /* 間隔の開始時刻と終了時刻のリスト */
Weekly Mask	構造体 /* 曜日毎の間隔のリスト */
Scheduler Name	文字列 /* 外部スケジューラ名 */
Destination	文字列 /* 宛先の運用エンティティ名称 */
Back_Up Destination_List	構造体 /* バックアップの宛先のリスト */
Active Destination (Read Only)	文字列 /* バックアップ中の運用エンティティ名称 */
Confirmed_Mode	整数 /* イベント報告が確認型か否か */

(2) ローカルイベント転送開始要求APIの関数形

Local Event Forward Start (User_ID, Req_ID, Sync_Mode, Discriminator_Construct)

関数パラメータ	構造体はポイントで指定する
User_ID	整数 / ユーザAPのID */
Req_ID	整数 / 要求ID */
Sync_Mode	整数 /* 要求の回答を待つ(同期)か非同期でもらうか */
Discriminator_Construct	構造体 / 選別器コンストラクト、フィルタ構造体 */

注) パラメータ名や属性名の先頭の*は必ず指定しなければならないものを示す。FDN(Local_ID)と Superior_FDN(Superior_Local_ID)は、どちらか一方を指定する。CMISレベルでEFDのMO生成要求を出すに必要なMOクラスと名前結合については、基本管理アプリケーションで自動設定する。

図 3 API 規定の例

Fig. 3 Examples of API specifications.

これらの API を使用して障害管理 AP 等の試作を進めた結果、各管理 AP で必要な管理操作に対応する API だけを選択でき、パラメータの設定も容易になることが確認できた。その反面、API の種類が多くなり、基本管理 AP での処理の増加につながることも分かった。11 種類の SMF を実装した結果、SMF では特定のサービスは 30 種類弱しか定義されていないのに対し、API の数は 80 種類を超えた。ユーザ AP 開発時の使いやすさと基本管理 AP での処理の増加はトレードオフであるが、ユーザ AP 開発時に 30 弱のサービスを 80 強に使い分ける必要がなくなる利点は大きいと考える。

5.2 付加機能

定期属性値読み出し機能は、別途試作した CMIP/SNMP (Simple Network Management Protocol) 変換プロキシ¹³⁾を介して、障害管理 AP から SNMP エージェントを監視するのに非常に有効であった。また、ローカルイベントログ管理機能やネットワークトポロジ管理機能も障害原因の分析のための情報源として有効であることが確認できた。

ただし、トポロジ管理機能に関しては、ネットワークの利用者間のエンドツーエンド管理を実現するために、論理的な接続関係やアプリケーションレベルまで合わせて管理するのが望ましいと考える。また、様々な管理情報体系を統一的に管理でき、かつ新しい管理情報体系の取り込みも容易にできる機構も強化する必要がある。今回の試作では、4.3 節で述べた方法によって 2 種類の体系を統一的に扱うことができた。しかし管理情報体系の種類は今後も増えていくため、それらを容易に取り込み、それまでのトポロジ情報の管理法に影響を与えない機構を確立する必要がある。

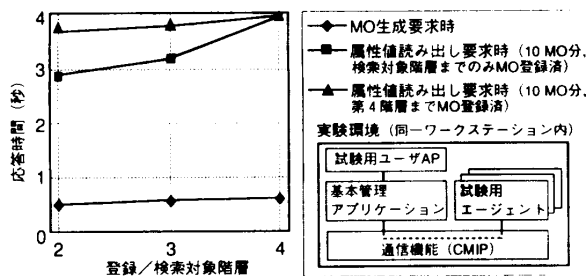
5.3 規模

試作したプログラムの C 言語レベルでの規模を表 4 に示す。共通機能としては、全体のプロセス管理や管理用の画面制御の他、試験用ユーザ AP (約 8K 行) を含んでいる。SMF 固有部分の 75% は転送情報の符

表 4 試作プログラムの規模 (C 言語での行数)

Table 4 Size of a prototype program (lines by C language).

機能	規模 [K 行]	
共通機能 (試験用ユーザ AP 等を含む)	37.9	
SMF 固有機能 (10164-1~8, 12, 13, 15)	21.0	
	符号化/復号関連	15.8
付加機能 (表 2 対応部分)	23.3	
合計	82.2	



注) ここで階層とはMOの包含木上の位置を意味する。実験では、各エージェントにおいて階層が一つ深くなるとMOの登録数が10倍になる(第2階層では10個、第4階層では1000個になる)ようにして実験した。また属性値読み出し要求は、IDのリスト形式で異なるエージェントにある10個のMOを指定して実験した。

図4 MO生成要求時と属性値読み出し要求時の応答時間
Fig. 4 Response times for MO create requests or attribute value get requests.

号化と復号に要している。この点からは、SMFで扱う管理情報をGDMOベースで容易に扱えるツールの利用が有効であり、そのようなツールとの融合を検討する必要がある。付加機能の中で一番大きいのはローカルフィルタ機能で約5K行であった。付加機能を追加して動作させる仕組みはできあがっているので、今後の機能追加でもこの程度の規模に収まるものと考ええる。

5.4 性能

性能は、実システムのCMIPエージェントがなかったため、図4に示す実験環境で評価した。ここでは、MO生成要求と属性値読み出し要求の応答時間を調べることで、基本管理APでMOに整数のIDを付与して管理していることの影響を評価した結果について述べる。この点を評価した理由は、ほぼすべての処理に共通に参与する付加機能だからである。

試験用ユーザAPレベルでのMO生成要求と属性値読み出し要求の応答時間を図4に示す。MO生成要求の場合、階層によらず応答時間はほぼ一定で、IDを付与してMOの識別名を登録するのに要する時間はほぼ一定であることが分かった。一方、属性値読み出し要求の場合、応答時間は検索対象(読み出すMOの)階層の深さと登録済みのMO数につれて増加していく。増加分はIDを識別名に変換するためのID検索時間の増加分であり、検索の階層に関しては1段深くなるのにもない、1個のMOあたり約0.02秒遅くなる。しかし、属性値読み出し全体の動作時間から見ればこの検索時間の増加の割合は小さいので、ID付与による処理遅延の影響は問題ないと考ええる。

6. おわりに

本論文では、従来のようにSMFを管理APと一体化して実装するのではなく、SMFをミドルウェア(基本管理AP)として実装していくことを提案した。こ

のため、SMFの実装に際し、従来のようにGDMOの面から管理情報の扱いを容易にするための提案をするのではなく、ユーザAPでの処理を容易にするために提供する付加機能と、ユーザAPの開発者が使いやすいAPIの規定法を提案した。さらに拡張性の高いプログラム構成と、実装上重要な処理方法についても提案した。最後に試作結果に基づき有効性について考察した。

今後の課題としては、考察で述べたように、トポロジ管理における論理的な接続関係やアプリケーションレベルの管理、および複数の管理情報体系のトポロジ情報を統一的に管理していくための汎用的な機構の確立がある。また、基本管理APへの新たなSMFの組み込みを容易にするため、GDMOツールとの融合も検討する必要がある。さらに、様々なユーザAPの開発と実環境への適用を通し、APIの使いやすさや付加機能の過不足についてさらに評価し、改善していくことも必要である。

参考文献

- 1) ISO/IEC 9596: Common Management Information Protocol Specification (1991).
- 2) ISO/IEC 9595: Common Management Information Service Definition (1991).
- 3) ISO/IEC 10165-4: Structure of Management Information - Part 4: Guidelines for the Definition of Managed Objects (1992).
- 4) ISO/IEC 10164-1~: Systems Management - Part 1~(1991~). (逐次標準化中)
- 5) Yamagishi, K., Sasaki, N. and Morino, K.: An Implementation of a TMN-Based SDH Management System in Japan, *IEEE Comm. Mag.*, Vol.33, No.3, pp.80-85 (1995).
- 6) 武藤, 有馬, 井藤, 渡部: OSI管理プラットフォームによるNMS開発環境, 信学技報, CS 95-73 (1995).
- 7) 堀内, 小花, 西山, 杉山, 鈴木: OSI管理のシステム管理機能(SMF)のためのプログラム開発支援ツールの実装, 信学会春季大会(1993).
- 8) 大鐘: TCP/IPとOSIネットワーク管理, 第5~6章, ソフト・リサーチ・センタ(1993).
- 9) 菅原: ネットワーク用エキスパートシステム, 人工知能学会誌, Vol.9, No.1, pp.40-47 (1994).
- 10) Network Management Forum: Reconfigurable Circuit Service: Configuration Management Ensemble, Forum 017 (1992).
- 11) Network Management Forum: LAN Fault Management Ensemble, Draft Ver.4.0 (1995).
- 12) 池田, 花木, 西田, 藤田: 網レベル故障管理法の一考察, 1995年信学会春季大会, B867 (1995).

- 13) Nakamura, N., Kashimura, N. and Motomura, K.: CMIP to SNMP Translation Technique Based on Rule Description, *Proc. ICCCN'95*, pp.266-271 (1995).

(平成7年9月25日受付)

(平成8年3月12日採録)



本村 公太 (正会員)

1981年大阪府立大学工学部電子工学科卒業。1983年同大大学院博士前期課程了。同年日本電信電話公社(現NTT)入社。以来、MHS準拠の電子メールシステム、高速・高機能トランスポートプロトコル、ATM-LAN、ネットワーク管理システムの研究開発に従事。現在、同社通信網研究所ネットワークインテグレーション研究部主任研究員。



平木久美子 (正会員)

1989年筑波大学第1学群自然科学類(地球科学)卒業。1991年同大大学院修士課程了。同年NTT入社。以来、ATM-LAN、ネットワーク管理システムの研究開発に従事。現在、同社通信網研究所ネットワークインテグレーション研究部研究主任。電子情報通信学会会員。



高野 誠

1984年神戸大学工学部システム工学科卒業。1986年同大大学院修士課程了。同年NTT入社。以来、分散交換システム、ATM-LAN、ネットワーク管理システムの研究開発、電話網のトラヒック管理業務等に従事。現在、同社サービス生産本部ネットワーク部担当課長。博士(工学)。平成5年度電子情報通信学会学術奨励賞授賞。計測自動制御学会、電子情報通信学会、日本ファジィ学会各会員。