

OSI 管理情報支援ツールMINT

中川路 哲男 宮内 直人 勝山 光太郎 水野 忠則
三菱電機(株) 情報電子研究所

OSI (開放型システム間相互接続) におけるネットワーク管理は、管理情報と管理プロトコルの両面から標準化が行なわれている。管理情報の定義のためには、テンプレートと呼ばれる記法が標準化されている。我々は、管理情報に関するソフトウェアの開発を支援するために、テンプレートによる管理情報定義から、管理情報へのアクセスを実現するソフトウェアを生成するツールMINT(Management Information support Tool)を開発した。MINTを使用することにより、適用領域毎に定まる管理情報に応じたネットワーク管理を実現するソフトウェアを効率的に開発することが可能となる。本稿では、ネットワーク管理ソフトウェア及びその効率的な開発のために開発したMINTの設計方針、機能、構成について報告する。

OSI Management Information Support Tool "MINT"

Tetsuo Nakakawaji, Naoto Miyuchi, Kotaro Katsuyama, Tadanori Mizuno

Information Systems and Electronics Development Laboratory

MITSUBISHI ELECTRIC CORPORATION

5-1-1 OFUNA KAMAKURA-CITY KANAGAWA 247 JAPAN

In the standardization of OSI(Open Systems Interconnection), "Network Management" is studied from management information aspect and management protocol aspect. For definition of management information, "template" notation is standardized. We have developed the tool "MINT" (Management Information support Tool) which generates software to access management information from definition of management information by the template notation. By using MINT, software for each network management application can be developed efficiently. In this paper, we present the design criteria, functions and software structure of management application software and MINT.

1. はじめに

ISO-IEC/JTC1及びCCITTでは、異機種システム間での相互接続を可能とする為に、OSI（開放型システム間相互接続）の標準化を行なっている。OSIの需要の高まりと共に標準化も順調に進捗し、各地で接続実験が行なわれる状況である。各種のシステムをネットワークに接続して運用するようになると、ネットワークも大規模化し、その管理が重要な課題となってくる。

OSIの応用サービスの一つとして、安全で且つコストにあった品質の通信を利用者に提供するために、ネットワーク管理（OSI管理）の標準化が進められている。OSIによる異機種間の接続を実用的なものとするためには、単に異機種システム間が接続できるだけでなく、利用者の要件に応じたネットワーク管理が不可欠であり、ネットワーク管理の標準化は急ピッチで行なわれている。ネットワーク管理の標準化は、管理情報とそれを転送する管理プロトコルの二つの側面から行なわれている。管理プロトコルに関しては、共通部分が既に国際規格^[1]となっており、かなり安定したものとなっている。

一方管理情報は、管理されるシステムやネットワークの用途などにも依存するため、その標準化も難しく、各層で管理される情報の候補を洗い出している段階である。管理システム・被管理システムの構築においては、運用されるネットワーク管理領域で定義された管理情報に応じて、対応す

表1 共通管理サービス

種別	概要
M-CREATE	マネージャからエージェントへ管理対象の生成を要求する
M-DELETE	マネージャからエージェントへ管理対象の消去を要求する
M-GET	マネージャからエージェントへ管理情報（属性）の獲得を要求する
M-SET	マネージャからエージェントへ管理情報（属性）の設定を要求する
M-EVENT-REPORT	エージェントからマネージャへ事象を報告する
M-ACTION	マネージャからエージェントへ動作を指示する

る管理情報を収集するソフトウェアを構築することが必要になるため、そのソフトウェアを効率的に開発することが重要である。

本稿では、ネットワーク管理ソフトウェア及びその効率的な開発のために開発したMINT(Management INFORMATION support Tool)の設計方針、機能、構成について報告する。

2. 管理情報とテンプレート

ネットワーク管理は、管理されるシステム（エージェント）側で管理対象毎に管理情報を収集してMIB（管理情報ベース）に蓄積し、管理するシステム（マネージャ）との間で、管理プロトコルを使用してその内容を交換することにより行なわれる。ネットワーク管理のモデルを図1に示す。

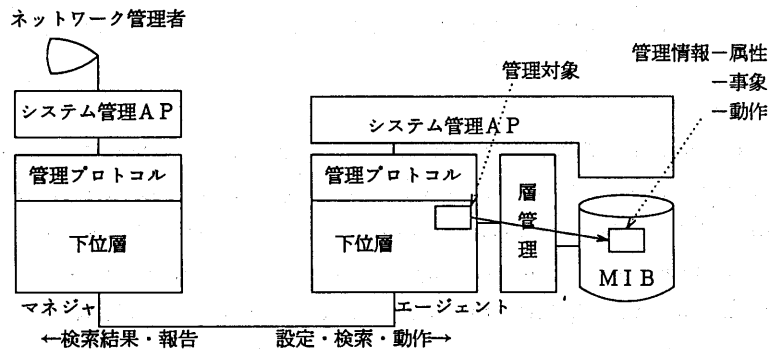


図1 ネットワーク管理のモデル

管理情報には、属性、事象、動作の三種類がある。管理情報の交換とは、管理情報へのアクセスであり、共通管理情報サービスでは表1に示す六種類のサービスを規定している。

例えばトランスポートコネクションを管理対象とすると、管理情報の属性には送信したPDU数、事象には再送回数の閾値超過、動作にはコネクション強制解放データの送信などが考えられる。この管理情報へのアクセスには、属性の読み出しとしては送信したPDU数の獲得、マネージャへの事象の報告としては再送回数の閾値超過の報告、エージェントへの動作としてはコネクション強制解放データの送信が考えられる。

表2 テンプレートの種類と規定内容

種類	規定内容(参照テンプレート)
管理対象クラス	管理対象名 上位管理対象クラス アロモーフジック管理対象クラス 管理対象の振舞い →振舞いテンプレート 管理対象の持つ属性 →属性テンプレート 管理対象への操作 =生成 =消去 動作→動作テンプレート 管理対象からの通知 →通知テンプレート オプションな管理情報 →条件付パッケージ
振舞い	振舞いの内容(自然言語orFDT)
属性	属性の継承関係 構文 (ASN. 1) 一致規則 (フィルタ操作) 属性の振舞い →振舞いテンプレート
動作	動作の内容 →振舞いテンプレート マネージャからの要求構文 マネージャへの応答構文
通知	通知の内容 →振舞いテンプレート マネージャへの要求構文 マネージャからの応答構文
名前づけ	上位管理対象下における相対 識別名の属性
特定エラー	管理情報における誤り
グループ属性	グループ化した属性
条件付パッケージ	ある条件下の管理情報の規定 (規定方法は管理対象クラス テンプレートに同じ)

このように、管理情報は管理対象毎に規定されるものであり、管理対象によって多種多様のものが想定される。また、管理対象自体も、そのシステムの規模や目的、運用されるネットワーク管理領域の用途によってかなり異なる。そのため、管理情報の体系や、汎用的な定義記法(テンプレート)がISOで検討されている^[2]。テンプレートは、その管理対象が持つ属性の内容、生成・消去の条件、通知・動作などの振舞いの内容、他の管理対象との関係と、管理対象の名前、各属性、通知、動作、振舞いの詳細を定義するための記法である。属性や通知・動作における引数の情報形式の規定には、OSIで標準化された抽象構文記法ASN. 1^[3]が使用されている。テンプレートの種類とその概要を表2に示す。また、テンプレートの例を図2に示す。

オブジェクトクラステンプレート

```
ExampleObjectClass MANAGED OBJECT CLASS
DERIVED FROM ISO/IEC 10165-2:top
CHARACTERIZED BY:
  BEHAVIOUR DEFINITIONS CommunicationErrorBehaviour;
  ATTRIBUTES QOS-Error-Cause GET;
  OPERATIONS CREATE with-automatic-instance-naming;
  DELETE deletes-contained-objects;
  NOTIFICATIONS CommunicationError;
REGISTERED AS {ObjectClass 1};
```

属性テンプレート

```
QOS-Error-Cause ATTRIBUTE
WITH ATTRIBUTE SYNTAX AttributeModule.QOSErrorCause;
MATCHES FOR Equality;
BEHAVIOUR QOSErrorBehaviour;
REGISTERED AS {AttributeID 2};
```

振舞いテンプレート

```
CommunicationErrorBehaviour BEHAVIOUR
DEFINED AS The CommunicationError notification is generated by ...
```

通知テンプレート

```
CommunicationError NOTIFICATION
BEHAVIOUR CommunicationErrorBehaviour;
WITH DATA SYNTAX EventModule.ErrorInfo;
WITH RESULT SYNTAX EventModule.ErrorResult;
```

関連ASN. 1モジュール

```
AttributeModule.DEFINITIONS ::= BEGIN
QOSErrorCause ::= ENUMERATED {
  responseTimeExcessive(0), queueSizeExceeded(1),
  bandwidthReduced(2), retransmissionRateExcessive(3)}
END
EventModule.DEFINITIONS ::= BEGIN
ErrorInfo ::= SET {
  [0] ProbableCause OPTIONAL,
  :
  }
END
```

図2 テンプレートの例

テンプレートによる管理情報の定義の内、エンティティやコネクションなど一般的なものはOSI管理の標準の一部として、標準化されつつある。各プロトコルに固有のものはプロトコル規格の一部として標準化される計画はあるが、その多くが未だ定義されていない。また、標準化される管理情報はスーパーセットであり、それら全てを収集することは本来の通信処理のオーバーヘッドとなる可能性がある。即ち、実際の運用においては、その目的に応じた管理のために必要な情報のみを絞り込んで収集する必要がある。これより、プロトコルが一旦実装すれば他の機種や他のアプリケーションでもほとんどそのまま流用可能であるのに比べて、管理情報の実装はケースバイケースで行なう必要があると言うことができる。

3. SMAPソフトウェアの設計

ネットワーク管理におけるマネージャは、管理ドメインの中で唯一存在し、専用の構築が可能であるのに対して、エージェントは多種多様の機種である。即ち、エージェントにおけるネットワーク管理のためのソフトウェア（システム管理アプリケーション（SMAP）ソフトウェア）は、機種、あるいは管理目的毎に開発する必要がある。我々は、SMAPソフトウェア開発の機会が多であることを想定して、その開発効率の向上を目的としてソフトウェアの設計を行なった。

エージェントにおけるSMAPソフトウェアの設計に当たっては、以下の設計方針を設定した。

- 1) 機種、管理目的、管理対象に依存しない、汎用的なソフトウェア構成。
- 2) 管理情報に依存する部分のテンプレートによる定義からの直接導出。これにより、基本的には、管理情報に依存しないモジュールと管理情報から導出されたモジュールを合体すればS

MAPを構築することが可能となるようにすること。

3) 高性能な管理操作。特に、層管理における管理情報の収集（設定）、マネージャからの管理情報の取得と、マネージャへの事象報告は、マネージャからの管理情報の設定、マネージャからの動作指示などに比べて即時性が要求されるので、この三つの操作の性能を重視した処理方式とすること。

4) オブジェクト指向アプローチの採用。汎用的なMIBを構築するために国際標準における管理情報のモデリングに採用されているオブジェクト指向の手法の良さを活かし、MIBをオブジェクト指向データベースとして実現すること。

1) の設計方針のためには、管理情報に依存する部分と依存しない部分の独立性を高める必要がある。我々は、SMAPソフトウェアを表3に示すモジュール構成で構築することとした。MIBへのアクセスを提供するモジュールは、4) の設計方針から、オブジェクト指向のインタフェース、即ち、オブジェクトの生成/消去、オブジェクトへのメッセージ送信という形で、管理情報へのアクセスを提供することとした。使用する言語としては、移植性・性能に優れるオブジェクト指向言語superC^[4]を採用した。また、3) の高性能を実現するために、層管理からでも、マネージャからの要求を実行する管理プロトコル処理モジュールからでも、MIBに高速にアクセスする必要がある。そのために、MIBへのアクセスを提供するモジュールを、層管理、管理プロトコル処理モジュールからリンクされるライブラリとして実現することとした。図3にSMAPソフトウェアのソフトウェア構成を示す。

表3 SMAPソフトウェアの構成

モジュール名	処理内容	管理情報への依存度
管理プロトコル処理	マネージャとの間で交換されるPDUの組立・分解（特定管理機能も含む）	無
管理対象木管理	管理対象木（包含木）の管理	有
MIBアクセス	管理情報の蓄積及びそれに対するアクセスインタフェースの提供	有

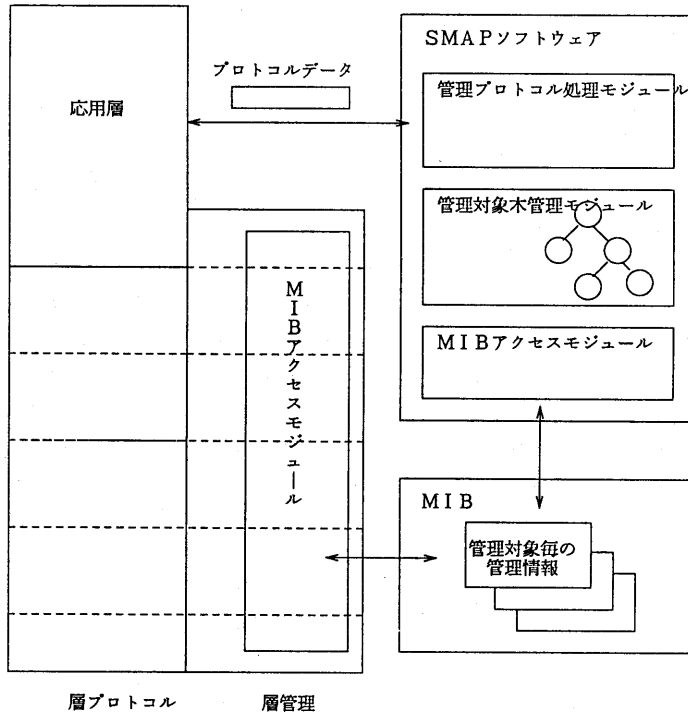


図3 SMAPソフトウェアの構成

4. MINT

3. で述べたソフトウェア構成において、MIBへのアクセスを提供するモジュール及び、管理対象木を管理するモジュールは、管理情報に依存する部分である。MIBへのアクセスを提供するモジュールは、具体的には各管理対象に相当するオブジェクトのクラス定義群から成る。我々は、設計方針2)を満足するために、テンプレート定義からこれらのモジュールを自動的に生成するツールMINTを開発した。MINTを使用することにより、層管理及び管理プロトコル処理モジュールでは、管理対象をテンプレートで規定されたままの形で扱うことができ、その内部的な実現方法について意識する必要がなくなる。また、テンプレート定義の規格自体が早期(DP)のもので不安定であることを考慮し、MINTはテンプレートのシンタックス定義自体をyacc/lexを利用して解析することにして、テンプレートのシンタックスの変更・拡張にも柔軟に対応できるようにし

た。図4に、MINTとSMAPソフトウェアの関係を示す。以下、MINTの入力と出力の対応を項目毎に述べる。

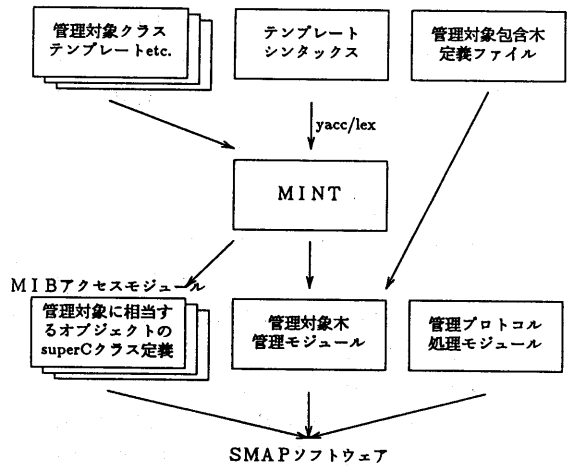


図4 MINTとSMAPソフトウェアの関係

4.1 MIBアクセスモジュール用クラス定義

4.1.1 管理対象名

MINTでは、管理対象毎にsuperCのクラス定義を出力する。その意味で、管理対象クラステンプレートとsuperCクラス定義が一對一に対応する。MINTは、管理対象クラステンプレートの中の管理対象クラスラベルを、superCクラス定義のクラス名として出力する。

4.1.2 上位管理対象名

管理対象クラステンプレートにおける上位クラスラベルは、その管理対象が属性や通知などの管理情報を継承する上位管理対象名である。MINTはこれをそのまま、superCクラス定義におけるスーパークラス名として出力する。管理対象クラステンプレートでは、複数の上位クラスラベルを記述することが可能であるが、superCでは多重継承をサポートしていないため、MINTでは一つしか記述することを許していない。この点については今後の拡張項目である。

また、管理対象クラステンプレートでは、アロモーフイズム管理対象を指定することにより、他の管理対象クラスとしても動作することを可能としている。この機構を用いることにより、例えばトランスポートコネクションのサブクラスであるクラス0トランスポートコネクションのインスタンスも、トランスポートコネクションのインスタンスとして扱うことができる。この機構もsuperCではサポートしていないが、これはオブジェクトクラス属性の値の一つとして、その管理対象クラスの値を持たせることにより解決した。

4.1.3 管理情報（インスタンス変数）

管理対象クラスのインスタンス変数は、管理情報そのものである。その意味で、保持している属性の型と値、発生している事象、指示されている動作をインスタンス変数に持たせる必要がある。

属性に関しては、管理対象クラステンプレートにおける属性ラベルに記述されている属性から属性テンプレートを検索し、属性の型に応じた型を持つデータをインスタンス変数として出力する。

属性の構文はASN. 1で定義されるので、ASN. 1の構文をデータとしてそのまま扱うことのできるツールAPRICOT^[5]を利用し、APRICOTプリコンパイラの出力であるデータ型定義テーブルへのポインタを、インスタンス変数として出力することとした。

事象報告においては、ディスクリミネータによってシステム内部の事象を属性としてフィルタにかけ、マネージャに必要な事象のみを報告する機構が用意されている。この機構を実現するために、管理対象クラステンプレートで定義された属性ラベルだけでなく、事象報告のために必要な属性もインスタンス変数として出力することとした。

動作に関しては、直接オブジェクトに対するメッセージとして実現する方法もあるが、動作の指示がマネージャから来ること及びその実行は層プロトコルに影響がある（コネクションのリセットなど）ことを考慮し、管理プロトコル処理モジュール側でマネージャから指示された動作内容をインスタンス変数にセットしておき、層管理モジュール側でそれを読み出して実行する方式を採用した。即ち、MINTは、管理対象クラステンプレートの動作ラベルに記述されている動作の型や引数に関する情報を動作テンプレートから検索し、インスタンス変数として出力する。

4.1.4 管理対象インスタンスの生成/消去

管理対象インスタンスの生成/消去を、オブジェクトのメソッドとして実現した。生成メソッドの中では、以下の処理を実行する。

- ・初期属性値の設定
- ・管理対象クラステンプレートの生成ラベルで参照オブジェクトが指定された場合の、属性値のコピー
- ・上位クラスインスタンスの生成と初期属性値の設定
- ・オブジェクト生成通知のための事象報告属性の設定

また、消去メソッドの中では、以下の処理を実行する。

- ・上位クラスインスタンスの消去

- ・オブジェクト削除通知のための事象報告属性の設定
- ・自オブジェクトの消去

4. 1. 5 管理情報の獲得

管理情報の獲得を、オブジェクトのメソッドとして実現した。本メソッドの中では、指定された属性や事象、動作の型に応じたインスタンス変数にアクセスし、その値を返す。フィルタが指定された場合には、そのフィルタを分解して、適合したものの値を返す。

4. 1. 6 管理情報の設定と事象報告

管理情報の設定を、オブジェクトのメソッドとして実現した。本メソッドの中では、指定された属性や事象、動作の型に応じたインスタンス変数にアクセスし、その値を設定する。属性値の設定に当たっては、管理対象クラステンプレートにおける属性ラベルのプロパティリストを検索し、その属性が書き込み可かどうかをチェックする。

また、閾値超過のように、属性値の設定の結果として事象報告が発生することがある。この場合、振舞いテンプレートに、その属性と事象報告の関係が記述されるが、振舞いテンプレートが自然言語記述であるため、その関係を機械処理することが困難である。我々は、振舞いテンプレートに簡単なシンタックスを追加し、属性と事象報告の関係などを規定できるようにした。

また、その処理においては、属性値の設定は層管理モジュール側で、事象報告は管理プロトコル処理モジュール側で行なう必要がある。これを実現するために、層管理モジュールで属性値を設定すると同時に振舞いテンプレートによる条件をチェックし、事象報告の必要があれば、それをインスタンス変数である事象属性値に反映させることとした。管理プロトコル処理モジュール側では、層管理モジュールから事象報告の発生を受けると、そのインスタンス変数にアクセスし、ディスクリミネータによるフィルタリングの後、マネージャに事象報告を行なう。このように、属性値の設定の延長上での事象報告処理を、管理情報の設定メソッド

の中で出力させることを可能とした。

4. 2 管理対象木管理モジュール

管理対象木管理モジュールは、管理対象間の包含関係を管理するモジュールである。本モジュールの機能を以下に示す。

- ・実際に生成された管理対象オブジェクト（インスタンス）間の包含関係の管理
- ・管理対象の自動名前づけ
- ・M-SET/M-GET/M-ACTION/M-DELETEにおけるスコープ（範囲指定）の分解

本モジュールは、管理対象包含木の定義ファイルと、管理対象クラステンプレート、名前づけテンプレートから、生成される。以上のMINTにおける入力テンプレートと出力クラス定義の関係を図5に示す。

なお、特定エラーテンプレート、グループ属性テンプレート、条件つきパッケージテンプレートは、機能が不明確な点や機械処理困難な点もあり、MINTの支援の対象とはしていない。

5. おわりに

管理情報を支援するツールMINTについて報告した。MINTを用いることにより、管理情報の定義に応じたネットワーク管理のためのソフトウェアを、効率的に開発することが可能となる。また、オブジェクト指向言語によるソフトウェアを出力することで、OSI管理のモデリングにおけるオブジェクト指向の概念をそのまま実際のソフトウェアとして具現化することが可能である。

今後は、具体的にプロトコルソフトウェアと結合するなどして、その汎用性・性能などの実用性を検証していく予定である。

<参考文献>

- [1] ISO 9595-9596: Common Management Information Service/Protocol (1990).
- [2] ISO DP 10165-4: Structure of Management Information Part 4: Guidelines for the Definition of Managed Objects (1989).
- [3] ISO 8824-8825: Abstract Syntax Notation

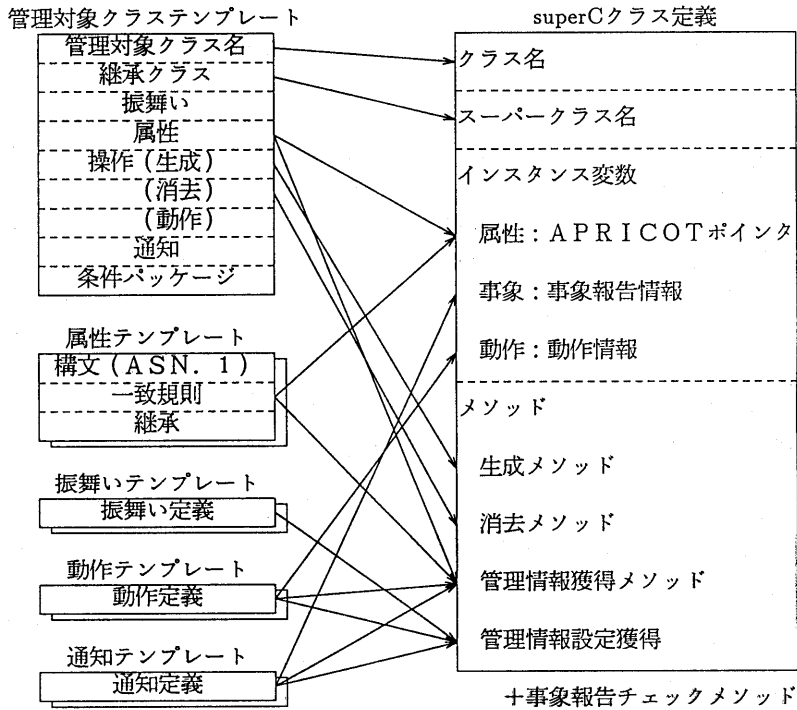


図5 MINTにおける入力テンプレートと出力クラス定義の関係

One(ASN.1) (1988).

[4] 勝山、佐藤、中川路、水野: 通信ソフトウェア向けオブジェクト指向言語superC, 情報処理学会論文誌, Vol.30, No.2, pp.234-241 (1989).

[5] 中川路、勝山、宮内、水野: OSI抽象構文記法支援ソフトウェアAPRICOTの開発と評価, 電子情報通信学会論文誌(D-I), J73-D-I, No.2 (1990).