

MIB エミュレータ：TMN に基づく 網管理マネージャの適合性試験ツール

西山 智[†] 小野 智 弘[†]
小花 貞 夫[†] 鈴木 健 二[†]

本論文は電気通信管理網 (TMN) に基づく網管理システムのマネージャの適合性試験方式を提案し、さらに提案方式に基づくツールである MIB エミュレータの設計、実装および評価について述べる。本論文で提案する適合性試験方式は、OSI プロトコルの適合性試験方式の 1 つである分散試験方式をベースとしている。さらに分散試験方式を構成する上位テストの代わりに仮想上位テストという概念を提案した点に特徴がある。また、管理情報ベース (MIB) をデータモデルとするアクティブデータベースを用いて下位テストを実現する。提案方式に基づくツールである MIB エミュレータを実装し、実際に使用されている TMN システムのマネージャを用いて評価を行った。その結果、必要な水準 (種類) の適合性試験が体系的に実施でき、提案方式の有効性が確認できた。また、MIB での更新性能、イベント処理性能はそれぞれ 20 件/秒、60 件/秒であり、適合性試験には十分な性能であることを示した。

MIB Emulator: A Tool for Conformance Test of TMN-based Managers

SATOSHI NISHIYAMA,[†] CHIHIRO ONO,[†] SADA O OBANA[†]
and KENJI SUZUKI[†]

This paper proposes a conformance test method for managers based on telecommunication management network (TMN) and presents the design and the implementation of a tool, named MIB emulator, based on the proposed method. Firstly, we propose a conformance test method for manager implementations. This method is based on distributed test configuration, one of the conformance test configurations for OSI protocol entity. We introduce the concept of virtual upper tester (UT) instead of the conventional UT. We propose a dedicated and active database specialized for MIB to realize the lower tester (LT). Secondly, we present the design and the implementation of MIB emulator, a tool based on the proposed conformance test method. Finally, we demonstrate the usefulness of MIB emulator through adopting it to the conformance test of actual manager implementations.

1. はじめに

近年、ネットワークの発展にともない、通信サービスの品質を向上させるために網管理システムが重要になってきている。網管理システムは一般に、電気通信管理網 (TMN)¹⁾ や簡易ネットワーク管理プロトコル (SNMP)²⁾ 等の標準的な網管理アーキテクチャに基づいて構築されている。TMN は、主に通信キャリアの電話網や海底ケーブルといった伝送システム等の管理に用いられている。一方、SNMP はインターネットやイントラネットといった、TCP/IP ベースのネットワーク管理で広く用いられている。これらの網管理アーキ

テクチャは、OSI (開放型システム間相互接続)³⁾ 管理の概念を受け継いでいるという点に共通的な特徴がある。

OSI 管理の概念では、マネージャが各ネットワーク要素 (エージェント) に存在する管理オブジェクト (MO) の集合からなる管理情報ベース (MIB) と呼ばれる仮想的なデータベースを操作して、網管理を実現する。MIB のスキーマは、管理対象となるエージェントの種類によって定まる。

網管理システムのマネージャを開発する際には、使用する網管理プロトコルの適合性試験に加えて、MIB のスキーマに対する適合性試験が必要となる。Sugiyama らは、TMN に基づく網管理システムのエージェントについて適合性試験方式を提案した⁴⁾。一般にエージェントは受動的に振舞い、マネージャがエージェントの持

[†] 株式会社 KDD 研究所
KDD R&D Laboratories, Inc.

つほとんどの機能を起動できる．このため Sugiyama らの提案では対象となるエージェントにテストを接続し，そのテストが適合性試験を主導する．この方式は，OSI プロトコルの適合性試験の枠組みを定めた ITU-T (国際電気通信連合技術セクタ) 勧告 X.290⁵⁾ あるいは国際標準 ISO/IEC9646⁶⁾ で示されている 4 つの抽象試験方式のうちの遠隔試験方式をベースとする．

一方，マネージャの試験については，マネージャに接続するエージェントから起動できるマネージャの機能はイベント処理に限られるため，エージェントとして振る舞うテストのみではマネージャの適合性試験は行えない．したがって，Sugiyama らが示した遠隔試験方式に基づく適合性試験方式はマネージャには適用できない．

本論文では，TMN に基づくマネージャの適合性試験方式について提案するとともに，提案方式を実装したツールである MIB エミュレータの設計と実装について述べる．MIB エミュレータは TMN を対象に実装したが，提案方式は MIB を操作するという共通の概念に基づく SNMP や他の網管理アーキテクチャにも適用可能である．また，本論文では，実際の TMN に基づくマネージャの適合性試験に適用した結果についても述べる．本論文は以下の構成をとる．2 章は，適合性試験に関する関連研究を示す．3 章はマネージャのための適合性試験方式について提案する．4 章は，提案方式に基づく MIB エミュレータの設計と実装について述べる．最後に 5 章は，実際の 2 種類の TMN マネージャに対して適合性試験を行った結果を示す．

2. 関連研究

2.1 OSI プロトコルに対する適合性試験方式

ITU-T 勧告 X.290 や国際標準 ISO/IEC9646 は OSI プロトコルに対する適合性試験の枠組みを定めている．この枠組みでは，試験対象 (以下 IUT: Implementation Under Test と呼ぶ) が，目的とするプロトコル仕様に適合するか否かを試験する．

適合性試験は，IUT に試験装置を接続して行う．まず，プロトコル仕様と実装状態を示す PICS (Protocol Implementation Conformance Statement) と呼ばれる適合性宣言を基に試験項目の集合である試験スイートを決定する．この試験スイートは，試験目的に応じて基本相互接続試験，能力試験および振舞試験の 3 種類に分類される．基本相互接続試験は，相互接続のための基本的な機能を確認する試験である．能力試験と振舞試験は，それぞれ静的，動的な適合性を確認する試験である．試験システムには，下位テスト (LT) と

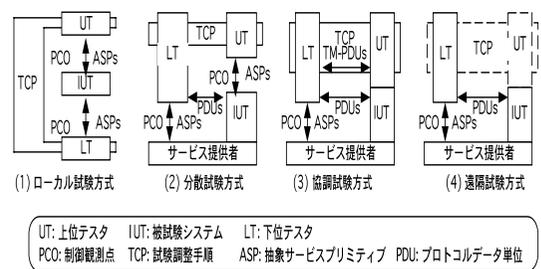


図 1 OSI プロトコルのための 4 つの適合性試験方式

Fig. 1 Four configurations for conformance test of OSI protocol entity.

上位テスト (UT) の 2 種類のテストが存在し，それぞれ IUT の上位および下位のサービス境界を制御し，結果を観察する．X.290 は図 1 に示すように，IUT, LT および UT の位置等に応じて，4 種類の適合性試験方式を定義している．図 1 において試験調整手順 (TCP) は試験における LT と UT の協調のための手順である．また制御観測点 (PCO) は試験の制御と観測が行われる点であり，PCO を経由して抽象サービスプリミティブ (ASP) が交換される．

2.2 TMN エージェントのための適合性試験方式

Sugiyama らの TMN エージェントに対する適合性試験方式⁴⁾は，まず TMN エージェントに対する適合性試験を OSI プロトコルの場合と同様に，基本相互接続試験，能力試験，振舞試験の 3 種類に分類した．この分類では，プロトコル仕様に相当する仕様として TMN での MIB スキーマを形式記述する GDMO (Guideline for Definition of Managed Object⁷⁾) を用いた．また PICS に相当するものとして MOCS (Managed Object Conformance Statement) および MRCS (Managed Relationship Conformance Statement) と呼ぶ適合性宣言を用いた．まず基本相互接続試験は，TMN で管理操作プロトコルとして用いられる CMIS (Common Management Information Service) 操作⁸⁾を，代表的な MO に対して代表的なパラメータで発行し，IUT での基本的な操作の受信，結果の送信，およびイベント発行を確認する．能力試験は，IUT が宣言した能力に準拠しているか否かを試験する．具体的には，MOCS および MRCS によって宣言された能力を基に，MO インスタンスに対して可能なすべての操作が発行でき，すべてのイベント処理が実行されることを確認する．最後に振舞試験は，適合性要件のすべての範囲にわたって可能な限り IUT を試験することであり，正しい操作に対する振舞いを確認する正常振舞試験と不正な操作に対する振舞いを確認する異常振舞試験に分かれる．

次に，Sugiyama らは TMN エージェントの適合性試験方式を提案した．この方式では，MOCS と MRCS から自動的に試験項目を抽出し，基本相互接続試験と能力試験を行う．また，試験シーケンスをシナリオとして記述することで正常振舞試験の一部も手動で実施可能とする．この方式は遠隔試験方式に基づいており LT が試験を主導する．すなわち，LT は試験項目に対する CMIS 操作を送出し，その結果を検査する．またイベントを受信した場合もその正当性を検査する．

3. TMN マネージャに対する適合性試験方式の提案

本章では，TMN を例にマネージャに対する適合性試験方式を提案する．まず，マネージャに対する適合性試験の枠組みを定義し，次に枠組みを構成する要素に必要な機能について述べる．

3.1 マネージャに対する適合性試験の枠組み

マネージャは，網管理者が発行したローカルな操作に基づき，エージェントに対して網管理操作を発行し，エージェントからの結果を編集して網管理者に通知する．またエージェントからのイベントを受信し，たとえば網管理者に通知する等の定められた処理を行う．マネージャが TMN での MIB スキーマの定義である GDMO 定義に適合しているかをチェックするには，IUT であるマネージャと，エージェントの代わりに接続した LT 間で交換される CMIS サービス（操作とその結果）を検査すればよい．しかしながら，エージェントとして振る舞う LT が起動できるマネージャ機能はイベント処理しかない．残りのマネージャ機能の適合性を確認するためにはマネージャに接続する上位テスト（UT）が必要である．したがって，マネージャに対する適合性試験は UT を持つ分散試験方式に基づく必要がある．ここでマネージャはシステム依存の操作と結果表示を網管理者に提供しているため，UT とのオンラインの汎用的なインタフェースはない．したがって，提案方式では UT の代わりに仮想 UT を設ける．仮想 UT は IUT とはオフラインのツールであり，適切な試験項目を試験者に提示する．試験項目に対応する IUT への操作の発行とその結果の確認は試験者が行う．

図 2 に仮想 UT と LT に必要な機能を示す．仮想 UT は仮想 PCO を実現するために必要な機能を持つ．一方，LT は LT に存在するイベント処理を対象とした PCO を実現する機能を持つ．さらに，LT は IUT と接続してエージェントとして振舞い，要求された操作に対する適切な（正しいあるいは不正な）結果を返

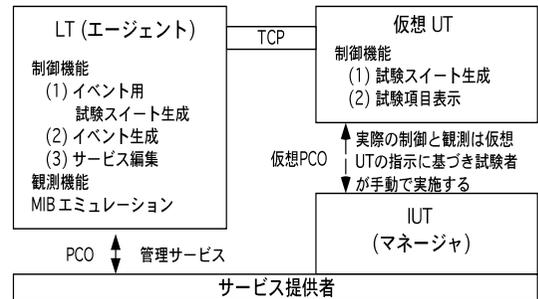


図 2 提案するマネージャのための適合性試験の枠組み
Fig. 2 Proposed framework for conformance tests of manager.

却する機能を持つ．提案方式ではこの機能を MIB エミュレーションと呼び，MIB 専用のデータベースによって実現する．本章の残りの部分では，これら 3 つの機能の詳細について述べる．なお，本論文では，適合性試験を Sugiyama らが提案した 3 種類（基本相互接続試験，能力試験，振舞試験）に分類して議論を進める．

3.2 仮想 UT における適合性試験の機能

仮想 UT が必要とする適合性試験機能は，仮想 PCO を実現する機能である．この PCO はイベント処理を除くすべてのマネージャ機能を担当する．

すでに述べたように，IUT であるマネージャは仮想 UT との間に汎用的なインタフェースもなければプロトコルもない．したがって，結果を仮想 UT が直接観察することはできないため，制御機能のみを仮想 PCO 機能として設ける．

3.2.1 制御機能

制御機能は試験スイート生成機能と試験項目表示機能からなる．

試験スイート生成機能は，要求された適合性試験種類（基本相互接続試験，能力試験，振舞試験）に応じて試験者用に試験スイートを生成する．具体的には基本相互接続試験では，管理操作ごとに 1 つずつ試験項目を割り当てる．能力試験では MIB スキーマに従い以下のような試験項目を作成する．

- MIB からすべての読み出し可能な値を読み出す．
- MIB に対して更新可能なすべての値を更新する．
- MIB に定義されたすべてのアクションを起動する．
- 複数の MO インスタンスを作成し，削除する．

最後に，正常振舞試験は IUT から一連の正しい操作を発行してその結果を確認する試験と考えられるが，正しい操作の組合せは無限にありうる．このため仮想 UT は正常振舞試験の試験スイートは作成せず，試験者の選択に委ねる．異常振舞試験については，仮想 UT

は以下の試験項目を含むような典型的な試験スイートを作成する。

- 結果に含まれるインスタンス名が不正である。
- 結果に存在しない属性や重複属性が含まれる。あるいは必須の属性が含まれていない。
- 結果に含まれる属性の ASN.1 構文が間違っている。

これらの試験の実施に必要な不正な結果応答は LT のサービス編集機能で生成できる。

試験項目表示機能は試験者に対して次に試験すべき項目を提示する。試験者はその試験項目と等価な操作を IUT に対して行い、IUT の振舞いを見て返される結果の正当性を確認する。

3.3 LT の適合性試験機能 (1) : PCO 機能

LT はイベント処理に関する PCO 機能を持ち、イベント処理に関する適合性試験を主導する。この機能は、制御機能と観測機能からなる。

3.3.1 制御機能

制御機能は、試験スイート生成機能、イベント生成機能、サービス編集機能からなる。

試験スイート生成機能は、仮想 UT の場合と同様に要求された適合性試験種類に応じてイベントのための試験スイートを生成する。仮想 UT では試験者用に試験スイートを生成するのに対し、LT ではイベントを生成するためのルール定義とそのルールを起動するためのシナリオを生成する。TMN でのルールおよびシナリオの詳細な設計は 4 章で述べる。

イベント生成機能は定義されたシナリオを逐次実行し、イベント処理に関する適合性試験を制御する。

最後に、サービス編集機能は、異常振舞試験を実施するために IUT に送出するサービス (操作結果とイベント通知) の内容を編集する機能である。この機能によりサービスの種別やパラメータを変更したり、送出タイミングを制御したりする。

3.3.2 観測機能

観測機能は LT と IUT 間でやりとりされるサービスが正しいかを確認する。やりとりされるサービスを試験者に表示するとともに、符号化の誤り等の文法誤りや、属性の型や値の誤り等の意味的な誤りをチェックする。試験者への表示は、試験者が意図したサービスが本当にやりとりされているかを確認するために用いる。

3.4 LT の適合性試験機能 (2) : MIB エミュレーション

提案方式では、仮想 UT は試験者に試験項目を順次提示するが、実際に IUT から発行される操作は試験項目から導かれるものであるとは限らない。たとえば、

IUT は仕様上、MO インスタンスの属性値を更新する前にその属性値を必ず読み出す仕様になっているかもしれない。また、試験者が提示されていない操作を実行する場合もある。したがって、操作要求が予測できないため、LT があらかじめそれに対する結果を用意しておくことは不可能である。さらに振舞試験で IUT の振舞いを確認するためには、一連の操作の結果は、たとえば更新した属性値は次に読んだ結果に反映されている等、前の操作結果を反映している必要がある。そこで、MIB の内容を LT に保持しておき、操作要求に対して常にそれを更新しておく。これにより要求された操作に対する正しい結果がツェに返却できる。我々はこの機能を MIB エミュレーションと呼ぶ。

MIB エミュレーションは要求された操作の分析、実行を行う機能に加えて、ルールの実行機能を持つ。通常 MIB 全体として内容が整合しているために MO インスタンスには他の MO インスタンスと何らかの関連が規定される場合がある。たとえば、あるコネクシオンの片方の 'TerminationPoint' が削除された場合、反対側の 'TerminationPoint' も削除される必要がある。通常このような関連は GDMO 中に自然言語により記述されている。提案方式では、このような関連を試験者がルールとして記述し、そのルールを実行することでインスタンス間の関連を維持する。また、MIB 内でのイベント生成規則としてもこのルール記述を使用する。ルール実行機能はこのルール記述を実行する機能である。ルールおよびその実行機能の設計の詳細については次章で述べる。

4. MIB エミュレータの実装と評価

本章では、TMN のマネージャを試験対象として提案した適合性試験方式を提供する試験ツールである、MIB エミュレータの設計と実装について述べる。

4.1 MIB エミュレータの機能

MIB エミュレータは、前章で述べた仮想 UT と LT を実現するための機能を提供する。また、適合性試験の効率化のために以下に示す機能を必要とする。表 1 にこれらの機能を要約する。

(1) シナリオエディタ :

試験者がシナリオを生成・編集したり、システムが生成したシナリオを編集したりする。

(2) ルールエディタ :

前述のルールを試験者が定義するために用いる。

(3) 適合性宣言エディタ :

試験者が、模擬対象のエージェントと IUT の適合性宣言を定義、編集する。

表 1 MIB エミュレータに必要な機能
Table 1 Requirements for MIB emulator.

要素	機能
仮想 UT (3.2 節)	制御機能 試験スイート生成 試験項目表示
LT の PCO 機能 (3.3 節)	制御機能 試験スイート生成 イベント生成 サービス編集 観測機能
LT の MIB エミュレーション (3.4 節)	操作要求分析 操作実行 ルール実行
補助機能 (4.1 節)	シナリオエディタ ルールエディタ 適合性宣言エディタ データ生成 データ読み込み

- (4) データ生成機能：
適合性試験に必要となる大量の MO インスタンスを自動生成する。
- (5) データ読み込み機能：
実際のエージェントの MIB の内容を取り込み MIB エミュレータ内の MIB に設定する。

4.2 MIB エミュレータの概略

MIB エミュレータは図 3 に示すように MIB プロセスと MIB 管理プロセスの 2 つのプロセスからなる。

MIB プロセスは、LT の機能の一部として MIB エミュレーションを行う。MIB 管理プロセスは、表 1 に示す残りの機能（仮想 UT と LT の PCO 機能）を実現する。また MIB プロセスの管理も行う。MIB プロセスと MIB 管理プロセスの間には MIB を操作するためと、MIB プロセスを制御するための 2 種類のローカルなサービス群を用いる。前者は基本的には CMIS サービスと同等であるが、GDMO 定義が禁止する操作も実行できる点が IUT からの CMIS サービスとは異なる。これは、たとえば、GDMO 定義では更新不可となっている属性の値を希望の値に設定するため等に必要なためである。

4.3 MIB プロセスの設計

4.3.1 MIB におけるデータ管理

MIB プロセスは、MIB 専用のデータベースである。そのデータモデルは MIB のモデルそのものであり、M-ACTION と M-EVENT-REPORT を除く CMIS 操作がデータ操作言語である。MIB プロセスは、拡張可能 DBMS ライブラリである ASSIST¹⁰⁾、ASN.1 コンパイラである Astool/C¹¹⁾、および GDMO コンパイラ¹²⁾を使用している。ASSIST はデータモデルに

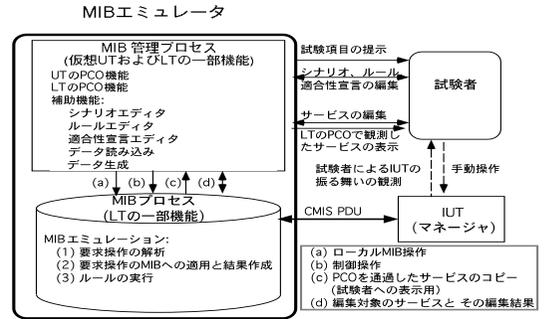


図 3 MIB エミュレータのプロセス構成と機能構成
Fig. 3 Processes of MIB emulator and their functions.

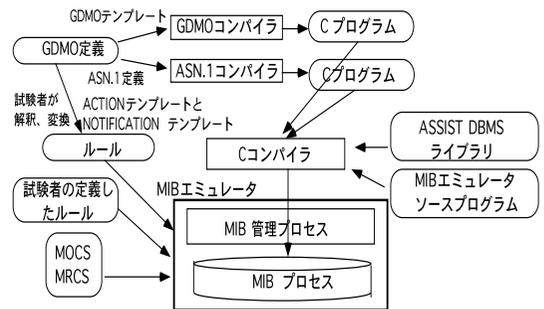


図 4 MIB エミュレータを特定のエージェントにカスタマイズする処理
Fig. 4 Customizing MIB emulator to specific agent.

依存しないレイヤ構造のデータベースライブラリであり、各種のアクセス手法、トランザクション処理機能、バッファリング機能等を提供する。我々は、CMIS 操作をデータ操作言語とし MIB のモデルを提供するレイヤ（MIB レイヤ）を ASSIST の最上位レイヤとして追加して、MIB プロセスを実現した。

エージェントの GDMO 定義は、MIB プロセスのデータベーススキーマに対応する。この GDMO 定義と、それに付随する ASN.1 定義はそれぞれ GDMO コンパイラと、ASN.1 コンパイラが C 言語に変換する。生成された C 言語のプログラムと、MIB レイヤを含む ASSIST ライブラリを合わせてコンパイルし、対象とするエージェントにカスタマイズした MIB プロセスを作る。図 4 にこのカスタマイズの流れを示す。

4.3.2 アクティブデータベースによるルール実行

MIB プロセスは、MO 間の関連を保ち MIB の内容の整合性を維持するためにルール実行機能を持つ。我々は MIB プロセスをアクティブデータベースとして実現し、ルール実行を可能とした。

ルールはイベント、条件、アクションの 3 つのフィールドからなる。イベントは、ルールが起動される原因を指定する。原因は、たとえば M-ACTION や M-SET

といった要求された CMIS 操作, 対象とする MO クラスおよび「MO インスタンスが生成された」「MO インスタンスがアクセスされた」「ある属性値が更新された」といった事象, の 3 つの組合せからなる。条件は, ルールが実行されるために必要な条件である。たとえば, 「インスタンス名が NE1 である」「ある属性値の値が 5 以上である」等が条件の例である。最後に, アクションはそのルールにより実行される内容である。アクションは M-ACTION を除く CMIS サービスの並びで指定する。

MIB プロセスは要求された, あるいはルールにより起動される CMIS 操作を実行するとき, すべてのルールのイベントと条件をチェックする。もしイベントと条件に合致するルールがあると, 元の操作とは独立にアクション部が指定する操作を実行する。もし指定されているサービスが M-EVENT-REPORT の場合, M-EVENT-REPORT を IUT に送出する。

4.4 MIB 管理プロセスの設計

MIB 管理プロセスは, LT と仮想 UT の制御および観測機能を GUI により提供し, 試験者と対話しながら適合性試験を実行する。適合性宣言エディタ, シナリオエディタ, ルールエディタやデータ生成, データ読み込みといった試験効率化のための補助機能についても MIB 管理プロセスが提供する。以下では, 試験者との対話部分を重点に述べる。

4.4.1 MIB スキーマの入力

MIB プロセスに対するローカル操作の入力やシナリオエディタ, ルールエディタ等でのシナリオやルールの編集では, 試験者は MIB スキーマに従って操作やそのパラメータ (たとえば属性の種類等) を入力する必要がある。MIB 管理プロセスは MIB プロセスが利用する GDMO 定義から C 言語に変換した MIB のスキーマ定義を同様に利用して, GUI により対話的に入力支援を行う。また ASN.1 値の入力についても, GUI により型やデフォルト値等の入力ガイドを行う ASN.1 インタプリタを利用して, 対話的に入力可能とする。なお, ローカル操作の入力では MIB スキーマに違反するパラメータの入力も可能とする。

4.4.2 適合性宣言の入力

適合性エディタでは入力を効率化するために, まず GDMO 定義から実装制限を盛り込まない適合性宣言の原案を自動生成し, IUT の MOCS や MRCS 等の適合性宣言と異なっている差分のみ入力させる。

4.4.3 試験スイートの生成

MIB 管理プロセスは試験スイートを現在 MIB に存在する MO インスタンスと, 入力された適合性宣言

表 2 適合性試験種類に応じた M-GET 操作に対する試験スイートの自動生成内容

Table 2 Test suites of specific conformance level for M-GET operation.

試験種類	対象とするインスタンス	属性
基本相互接続試験	各 MO クラスごとに 1 インスタンス	すべての属性
能力試験レベル 1	各 MO クラスごとに 1 インスタンス	すべての属性と個々の属性
能力試験レベル 2	基本的にすべての MO インスタンス。ただし, あるインスタンスの配下で同一クラスのインスタンスが複数ある場合は代表の 1 インスタンスのみ	すべての属性と個々の属性
能力試験レベル 3	すべての MO インスタンス	すべての属性と個々の属性

表 3 シナリオで使用できる文

Table 3 Supported statements in scenario.

カテゴリ	文	説明
ローカル操作	CREATE SET DELETE	MO インスタンスを作成 MO インスタンスの属性値の変更 MO インスタンスを削除
制御構造	BEGIN LOOP END LOOP SLEEP BEEP	ループを開始 ループを終了 指定時間待つ ビーブ音発生
マクロ	CREATE-PL	複数インスタンス生成用マクロ

```

-----
BEGIN LOOP(100, T1) .... ループの開始 (ID=T1)
SET(DN=(System:NE1),
  Values(OperationalState=Active),,,)
  .... 属性値の変更
SLEEP(2000) .... 2秒停止
BEEP .... ビーブ音発生
END LOOP(T1) .... ループの終了 (ID=T1)
-----

```

図 5 シナリオの例

Fig. 5 Example of scenario.

から自動生成する。この試験スイートの種類として, 基本相互接続試験, および能力試験レベル 1 から 3 の 4 種類を提供する。表 2 に M-GET 操作に関する各種種類ごとの試験スイート内容を示す。

4.4.4 シナリオ

MIB プロセスに対する一連のローカルな操作をシナリオとして登録することができる。シナリオを用いて, MIB プロセス内の MIB を更新したりそこからイベントを発生したりできる。シナリオにはループ制御といった基本的な制御構造が用意されており, 負荷試験等のために大量のイベントを発生させる等に使用できる。表 3 にシナリオで使用できる文を示す。図 5 にシナリオの例を示す。このシナリオは, 2 秒間隔で

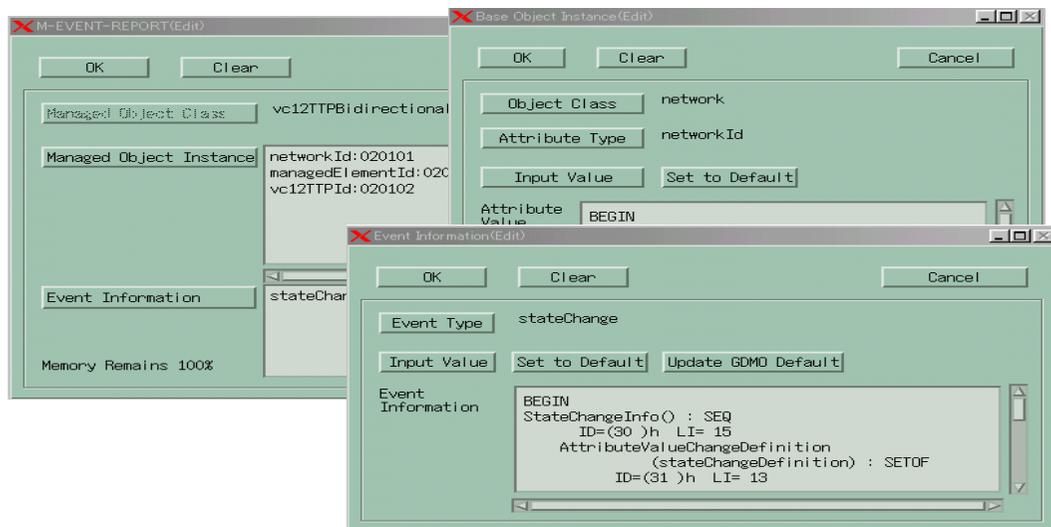


図 6 GUI の例：シナリオ編集画面

Fig.6 Example: scenario editing window.

100 回 M-SET 操作を発行する。

4.4.5 サービス編集機能

サービス編集機能は、試験者が MIB プロセスから IUT に送出するサービスを途中で編集する機能を提供する。まず MIB プロセスからサービスを受信すると試験者に通知し、編集を待つ。試験者が GUI により編集を行うと、編集後のサービスが MIB プロセス経由で IUT に送られる。この機能により、異常振舞試験に必要な不正な結果応答を作成できる。また、試験者が編集を遅らせることでタイムアウトのような一部のプロトコル試験も行える。

4.5 実装

MIB エミュレータは C 言語で実装されており Solaris OS 上で動作する。MIB エミュレータのプログラム規模は、GDMO 定義およびそれに含まれる ASN.1 定義からコンパイラが自動生成する C のコードを除くと、約 63 K 行である。下位のプロトコルスタックには、市販されている CMIP ソフトウェアを使用している。図 6 に GUI の例を示す。この例では、シナリオエディタで M-EVENT-REPORT 操作の詳細を指定している。

5. 評価

これまでに、表 4 に示す 2 種類の網管理システムで、マネージャの適合性試験を実施した。本論文では、システム 1 の場合について評価結果を示す。なお以下での性能値は SUN SPARC Server1000 (OS: Solaris2.5) 上での測定値である。

表 4 評価に用いた網管理システム

Table 4 Network management systems used for evaluation.

	システム 1	システム 2
エージェント名	SDH 網要素オペレーションシステム	デジタルクロスコネク
代表的な GDMO 定義	X.721, M.3100, Q.821, Q.822, 等	
MO クラス数	187	208
アクション定義数	26	20
イベント定義数	22	20

5.1 特定の網管理システムへのカスタマイズに関する評価

- (1) GDMO 定義によるスキーマのカスタマイズ：行数換算で GDMO 定義の 95%以上が GDMO コンパイラにより自動的に処理できた。生成された C プログラムは 137 K 行であった。GDMO 定義の残りの 5%は、ACTION および NOTIFICATION テンプレートであり自然言語によりその振舞いが記述されている。試験者がこれらのテンプレートを解釈して相当するルールを定義した。26 の ACTION と 22 の NOTIFICATION があり、約 6 時間でこれらをルールに書き換えられた。
- (2) 試験用 MO インスタンスの作成性能：データ生成機能およびデータ読み込み機能による MO インスタンスの作成性能は、約 20 インスタンス/秒であった。システム 1 の実エージェントは通常約 7,500 のインスタンス(クラス数は 13)を含む。MIB エミュレータは適合性試

表 5 システム 1 での M-GET 操作について自動生成した試験項目数

Table 5 Number of generated test cases for M-GET service to system 1.

適合性試験種類	試験項目数
基本相互接続試験	13
能力試験レベル 1	141
能力試験レベル 2	559
能力試験レベル 3	1.5×10^6

験を行うためにこれらのインスタンスを約 7 分で生成した。

以上、約 1 日程度の作業で特定の網管理システムにカスタマイズできた。

5.2 適合性試験に関する評価

(1) 試験スイートの自動生成機能について：

M-GET 操作に対する試験スイートの自動生成結果を表 5 に示す。能力試験レベル 3 では、試験項目数が 1.5×10^6 に達しており、仮想 UT からの指示を試験者が手で IUT に投入するには実際的ではない。表から、能力試験のレベル 1 あるいは 2 が試験スイートとして実際的であると考える。

(2) 実際のシステム開発試験との比較：

システム 1 の開発時の試験手順書を分析した結果、MIB スキーマに関する適合性試験として 99 項目の試験項目が該当した。これらのうち、87 項目は MIB エミュレータの基本相互接続試験および能力試験レベル 1 の一部の試験項目に該当しているが、逆に基本相互接続試験の項目すべてをカバーしているものではなかった。MIB エミュレータを用いることで抜き打ち的ではなく指定した種類の適合性試験が体系的に実施でき、適合性の確認がより高い精度で行えると考えられる。また残りの 12 項目については、有効な振舞試験に該当するものであり、IUT から相当する一連の操作を行い、MIB エミュレータ側で MIB エミュレーション機能により適切な結果を返すことで試験が実施できた。最後に異常な振舞試験については、実際のシステム開発試験では実際のエージェントが不正な結果応答を返すことが困難なため該当する試験項目がなかったが、MIB エミュレータでは不正な結果応答が作成できるため、エージェント側の異常動作といった実エージェントを用いる場合には困難な異常振舞試験が容易に行えた。この結果から、提案方式および MIB エミュレータはマネージャの適合性試験に非常に有効であると考

える。

5.3 安定化試験、負荷試験への適用について

MIB エミュレータは、イベント発生およびインスタンス作成がそれぞれ 60 件/秒、20 件/秒処理可能であり適合性試験には十分な性能を持つ。

網管理システムの試験では、適合性試験に加えて長期運用のための安定化試験や、大規模障害に備えた過負荷試験が必要となる。MIB エミュレータはシナリオを記述することで長期間にわたりイベントを自動的に送出することが可能であり、安定化試験のためのツールとして利用できる。また、複数の MIB エミュレータを同時に起動することで、短時間に多数のイベントを発生でき過負荷試験にも有効である。

6. ま と め

本論文では、網管理システムのマネージャに対する適合性試験方式を提案し、提案方式に基づくツールである MIB エミュレータの設計、実装について述べた。提案方式は分散試験法に基づいており従来の UT とその PCO の代わりに仮想的な UT と PCO を設けた点に特徴がある。提案方式に基づいて実装した MIB エミュレータは MIB プロセスと MIB 管理プロセスの 2 つのプロセスからなる。MIB プロセスは、専用のアクティブデータベースで LT 機能の一部である MIB のエミュレーション機能を実現する。MIB 管理プロセスは、LT と仮想 UT に必要な制御と観測機能を提供する。実際の TMN に基づく網管理システムにおけるマネージャに対する適合性試験に適用し、MIB エミュレータの実用性を確認した。MIB エミュレータは TMN に基づくマネージャを対象として実装したが、提案する適合性試験方式は、SNMP 等 MIB を操作する他の網管理アーキテクチャについても適用が可能である。

謝辞 日頃ご指導いただく(株)KDD 研究所村谷拓郎代表取締役所長に感謝いたします。

参 考 文 献

- 1) ITU-T: Telecommunication Management Network-Generic Network Information Model, ITU-T 勧告 M.3100 (1992).
- 2) IETF: A Simple Network Management Protocol, IETF Request For Comments 1157 (1990).
- 3) ITU-T: Management Framework Definition for Open Systems Interconnection for CCITT applications, ITU-T 勧告 X.700 (1992).
- 4) Sugiyama, K., et al.: Implementation and evaluation of MIB tester for OSI management,

Proc. IFIP/IEEE IM'97, pp.447-460 (1997).

- 5) ITU-T: OSI Conformance Testing Methodology and Framework for Protocol Recommendations for CCITT Applications-General Concepts, ITU-T 勧告 X.290 (1992).
- 6) ISO/IEC 9646: Information technology-Open Systems Interconnection-Conformance testing methodology and framework (1991).
- 7) ITU-T: Requirements and Guidelines for Implementation Conformance Statement Proformas associated with OSI Management, ITU-T 勧告 X.724 (1992).
- 8) ITU-T: Common Management Information Protocol Specification for CCITT applications, ITU-T 勧告 X.711 (1992).
- 9) McCarthy, D. and Dayal, U.: The Architecture of an Active Database Management System, *Proc. 1989 ACM SIGMOD Conference on Management of Data* (1989).
- 10) 西山 智ほか：拡張可能 DBMS 構築技法に基づく高速 OSI ディレクトリ用 DBMS の設計と評価, 情報処理学会論文誌, Vol.34, No.6, pp.1486-1496 (1993).
- 11) Hasegawa, T., et al.: Implementation and Evaluation of ASN.1 Compiler, *Journal of Information Processing*, Vol.15, No.2, pp.157-167 (1992).
- 12) 堀内浩規ほか：OSI 管理のためのシステム管理機能(SMF)スタブジェネレータの実装, 1993 年度電子情報通信学会春季全国大会予稿集, Vol.3, p.225 (1993).

(平成 11 年 4 月 28 日受付)

(平成 11 年 12 月 2 日採録)



西山 智 (正会員)

昭和 36 年生。昭和 59 年東京大学工学部電気工学科卒業。同年国際電信電話(現 KDD)(株)入社。平成 3 年米国テキサス大学オースチン校計算機科学科修士課程修了。現在、(株)KDD 研究所ネットワーク管理グループ主任研究員。この間、データベース、ネットワーク管理の研究に従事。平成 5 年度電子情報通信学会学術奨励賞受賞。電子情報通信学会会員。



小野 智弘 (正会員)

昭和 43 年生。平成 4 年慶應義塾大学工学部電気工学科卒業。平成 6 年同大学院計算機科学専攻修士課程修了。同年国際電信電話(現 KDD)(株)入社。現在、スタンフォード大学客員研究員。この間、ネットワーク管理、通信支援データベースの研究に従事。平成 8 年度電子情報通信学会学術奨励賞受賞。電子情報通信学会会員。



小花 貞夫 (正会員)

昭和 28 年生。昭和 51 年慶應義塾大学工学部電気工学科卒業。昭和 53 年同大学院修士課程修了。工学博士。同年国際電信電話(現 KDD)(株)入社。現在(株)KDD 研究所取締役。この間、パケット交換方式、ネットワークアーキテクチャ、OSI プロトコルの実装、データベース、ネットワーク管理の研究に従事。本会理事。電子情報通信学会会員。



鈴木 健二 (正会員)

昭和 20 年生。昭和 44 年早稲田大学理工学部電気通信学科卒業。昭和 44 年から 45 年までオランダのフィリップス国際工科大学に招待留学。昭和 51 年早稲田大学大学院博士課程修了。工学博士。同年国際電信電話(現 KDD)(株)入社。現在(株)KDD 研究所代表取締役副所長。この間、磁気記録、パケット交換、ネットワークアーキテクチャ、分散処理の研究に従事。昭和 62 年度前島賞、平成 4 年度電子情報通信学会業績賞、平成 7 年度科学技術庁長官賞受賞。平成 5 年度より電気通信大学大学院情報システム学研究科客員教授。平成 9 年度、10 年度本会理事。電子情報通信学会、IEEE 各会員。