

## ASN.1テストツールの実用性向上に関する機能拡張

4N-3

野村 真吾<sup>†</sup> 長谷川 亨<sup>†</sup> 堀内 浩規<sup>††</sup>

<sup>†</sup>国際電信電話株式会社 上福岡研究所

<sup>††</sup>(株) オーエスアイ・プラス

### 1. はじめに

OSIやISDN等のアプリケーションプロトコルでは、プロトコル要素(PDU)のデータ構造が標準記法ASN.1<sup>[1,2]</sup>を用いた抽象構文として定義され、そのデータ構造から転送するバイト列への符号化規則はASN.1で定められている。アプリケーションプロトコルの実装/試験を行うためには、ASN.1で定義したデータの処理を支援するツール<sup>[3-5]</sup>が重要であり、筆者らは試験用のツールとしてASN.1テストツールを開発してきた<sup>[3]</sup>。今回、テストツールを用いた試験作業の効率を向上させるため、ユーザインタフェース等の機能拡張を行った。本稿では、機能拡張の概要について報告する。

### 2. 機能拡張の方針

テストツール<sup>[3]</sup>の実用性を向上させるために、以下の方針で機能拡張を行った。

#### (1) ASN.1言語仕様

部分型、自己参照/相互参照によるデータ型定義を新しくサポートすることによりマクロを除くASN.1の言語仕様全てを処理の対象とする。

#### (2) ユーザインタフェース

通信プログラムの試験作業では、被試験プログラムの入力となるPDUの作成や、出力されるPDUの解析等の作業を支援するユーザインタフェースの機能が重要である。本ツールでは、使用者をガイドしながら新しいPDUを作成するジェネレータ、PDUを編集するエディタ<sup>[3]</sup>、PDUの解析結果の一覧を出力するプリティプリンタを提供する。

#### (3) 移植性

UNIX、VAX VMS、MS-DOS等の様々なOS上に実装するために、移植性を高める。

#### (4) 特定アプリケーションのための機能拡張

ASN.1ではマクロ定義記法を用いて新しい記法(マクロ)を定義することができるが、定義されたマクロの意味は形式的に規定されない。従って、マクロ定義を処理してマクロを自動的に展開することは困難であり、プロトコルの実装者/試験者が行う必要がある。ただし、ROSEプロトコル上に実装されるRPC(遠隔手続呼出し)系プロトコルのサービスを

規定するOPERATION等のRO記法<sup>[6]</sup>のマクロをサポートする。今後、標準化や実装が予定されるプロトコルの多くが、ROSE上に定義されると予想するためである。

### 3. テストツールの機能拡張

#### 3.1 テストツールの概要

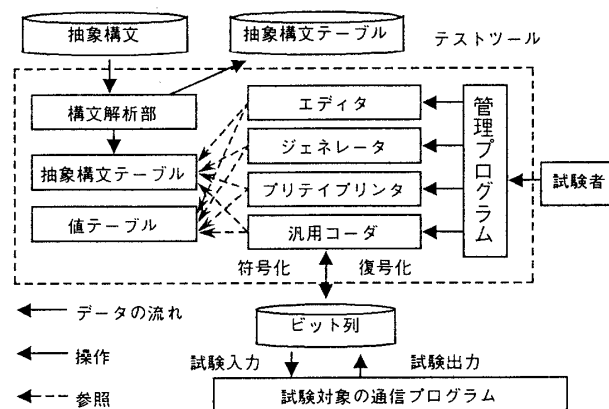


図1 ASN.1テストツール

テストツールは、図1に示す構成を持つ。ASN.1の抽象構文は内部的な構文木(抽象構文テーブル)により、実際の値も木構造(値テーブル)により表現する<sup>[3]</sup>。構文解析部が、抽象構文を抽象構文テーブルに変換し、各処理はこのテーブルを用いて処理を行う。試験作業で使用するPDUはビット列としてファイルに保持し、コードがこのビット列を復号し、値テーブルを生成してから各処理を行う。符号/復号化を実行するコードプログラムの実現には、ASN.1の型毎に専用のプログラムを生成する方式と、あらかじめ汎用のプログラムを準備する方式が考えられるが、符号/復号化速度より処理の柔軟性を重視して、後者の方式を採用している。

ツールは全てC言語で実装し、構文解析部はUnixのツールyaccを使用して実装した。各処理はモジュール化して実現されており、試験者はツール全体の管理を行うプログラムを介して、各処理を起動する。以下では、拡張した機能について述べる。

#### 3.2 ユーザインタフェース

先に開発したテストツールはPDUの編集を行うエディタのみを実現したが、試験入力となるPDU

Extension of ASN.1 Test Tool for Practical Use

<sup>†</sup>Shingo NOMURA, <sup>†</sup>Toru HASEGAWA, <sup>††</sup>Hiroki Horiuchi

<sup>†</sup>KDD Kamifukuoka R & D Laboratories <sup>††</sup> OSI Plus Corp.

を新規に作成したり、PDUを一通り解析する作業を効率的に行えるように、ジェネレータ、プリティプリンタ機能を新しく提供する。

#### (1) ジェネレータ

使用者をガイドしながら新しいPDUを作成するための機能を提供する。ジェネレータは抽象構文の情報に従って使用者に値の入力を促すため、使用者は抽象構文を意識することなく作業を進めることができる。図2にMHSプロトコルのO/RNameの値の作成例を示す。アンダラインの部分が使用者の入力である。最後の質問は、O/RNameを不定長か固定長のどちらで行うかを質問している。

```
>> MAKE ORName (originator)
>> MAKE StandardAttributeList
  Make "CountryName" ? (Y/N) Y
>> MAKE CountryName
  Choice
    1 : Numeric
    2 : Printable
    2
    "Printable" --> JPN
<< END CountryName
Length by (80)h ? (Y/N) N
Make "PersonalName" ? (Y/N) Y
>> MAKE PersonalName
  "SurName" --> HASEGAWA
  "GivenName" ? (Y/N) N
<< END PersonalName
Do you continue "StandardAttributeList" ? (Y/N) N
Length by (80)h ? (Y/N) N
....
```

図2 ジェネレータによる作成例

#### (2) プリティプリンタ

ビット列の復号結果を一覧できるように、文字列として出力する機能を提供する。表示方法としてはASN.1の値記法を採用する方法も考えられるが、試験時には、識別子オクテットや長さオクテットの実際の値の情報を知る必要があるため、値だけでなく、ビット列に関する情報も併せて表示する。ASN.1の構造型の入れ子に従って字下げをして表示し、文字列の検索、表示箇所の指定等ができるようにした。また表示結果をアスキーファイルとして、保存することも可能である。図3に、DirectoryプロトコルのオペレーションREADを運ぶROIvapduの表示例を示す。

```
BEGIN >> ROSEapdu() : CHOICE
BEGIN >> ROIvapdu(roiv-apdu) : SEQ
  ID = (A1)h LI = 339
  BEGIN >> invokedIDType(invokedID) : INT
    ID = (02)h LI = 1
    Value[ 1 ]
  END<< invokedIDType
  BEGIN >> OPERATION(operation-value) : INT
    ID = (02)h LI = 1
    Value[ 1 ]
  END<< OPERATION
  BEGIN >> ReadArgument(argument) : SET
    ID = (31)h LI = 225
    BEGIN >> Name(object) : CHOICE
      ....
```

図3 プリティプリンタによる表示例

#### 3.3 RO記法のサポート

OSIやISDNでRPC系のプロトコルは、ROSEやTCAPプロトコルを用いてリモートオペレーシ

ン(遠隔操作)を実現する。オペレーション等の引数や戻り値等のパラメータはOPERATION, ERROR, BIND, UNBINDマクロを用いて定義され(RO記法)、その値はROSEやTCAPのPDUとして転送する。そこで、このパラメータ及びROSE, TCAPのPDUのレベルでの試験作業を支援するために、以下の機能を実現した。

構文解析部の入力は上記のマクロに従った記述になり、その構文規則として、マクロ定義記法の構文規則でなく、OPERATION等のマクロの構文規則を採用した。エディタ等では、ROSEプリミティブのユーザデータとなるオペレーション等のパラメータ、ROSE PDUに対する編集等の試験作業ができるようにしている。図3では、ROSE PDUで転送されるオペレーションのパラメータを解析する例が示されている。また、コードではオペレーション値に応じて、パラメータを符号/復号化するための機能を追加している。

#### 3.4 移植性

各計算機、OSにより異なる点をカプセル化して定義することにより移植性を高めた。具体的には、ワードサイズ、バイトおよびビットの順序、ファイル入出力、端末が提供する画面の大きさを考慮して実装を行った。現在以下の計算機上で動作することを確認している。

NEWS, SUNワークステーション(UNIX) / VAX  
シリーズ(VMS) / PC-9800, J-3100(MS-DOS)

#### 4. おわりに

ASN.1テストツールのユーザインタフェースとして、ジェネレータ/プリティプリンタを追加することにより、PDU等の構造を意識することなくテストデータの作成/解析が可能となり作業効率が向上した。さらにRO記法のサポートを行うことにより、本ツールの提供する試験作業の適用範囲を広げることができた。また各処理をモジュール化して作成することで、拡張性を高めた。現在、プリティプリンタ機能を利用して、回線を流れるASN.1のビット列のデータをモニタする保守用パソコンを実現する検討を進めている。最後に日頃御指導頂くKDD上福岡研究所通信ソフトウェア研究室 小西室長、コンピュータ通信研究室加藤主査に感謝する。

#### 参考文献

- [1]:ISO, "ASN.1", ISO / IS 8824 / 8825.
- [2]:CCITT, Rec. X.208 / 209, Nov. 1987.
- [3]:長谷川, 野村, 堀内 "ASN.1支援ツールの開発 - コンパイラおよびエディタ-", 情報処理学会マルチメディアと分散処理研究会, 39-4, Sept. 1988.
- [4]:Nakakawaji, et. al., "Development and Evaluation of APRICOT", The second ISIIS, pp55-62, Nov. 1988.
- [5]:Ohara, et. al., "ASN.1 tools for Semi-automatic Implementations of OSI Application Layer Protocols, pp63-70, The second ISIIS, Nov. 1988.
- [6]:CCITT, Rec. X.220 / 226, Dec. 1987.