ASN.1データベースのための高速なASN.1処理系の設計

2T－4

西山 智　堀内 浩規　小野 智弘　小花 貞夫　鈴木 健二
国際電信電話株式会社 研究所

1. はじめに

抽象構文記法1(ASN.1)[1]は、OSI応用層で扱うプロトコルやデータ要素の情報を機種に依存することなく交換するための、データ型の記法と標準的な符号化規則を定めている。OSIディレクトリ[2]等OSIの応用によっては、ASN.1で定義されたデータ型を持つ情報をデータベースに格納する必要がある。筆者らは、現在、このためのデータベース(ASN.1データベース)の開発を進めている[3]。本稿では、ASN.1データベースで重要となる高速なASN.1処理系の設計について報告する。

2. 設計の基本方針

ASN.1データベースで使用するASN.1処理系は以下の要求条件を満たす必要がある[3]。

- ASN.1で定義されたデータベーススキーマの変更を可能とするため、処理するASN.1定義を動的に変更できること。
- データベースに格納する大規模なASN.1情報を高速に処理できること。
- インデックスを付与するために重要となる識別符号化規則(DER)が扱えること。

これまでに、いくつかのASN.1処理系が報告されている[4, 5]が、これらの要求条件を全て満たしているものはない。そこで、これらの要求条件を実現するために以下の基本方針でASN.1処理系を設計する。

- ASN.1定義を、ASN.1処理系の利用者プログラムの実行時には変更する必要がない部分(静的定義部)と、実行時に変更されうる部分(動的定義部)に分ける。高速化のために前者は、コンパイラが生成する符号化/復号関数が符号化/復号処理を行う。後者はインタプリタが符号化/復号処理を行う。
- 大規模なASN.1情報を高速に符号化/復号するため、符号化オクテット列の一部を復号する、あるいは逆に一部に既に符号化したオクテット列が混在する情報を符号化することを可能とする。
- 基本符号化規則(BER)に加えてDERを扱えるようにする。

3. ソフトウェア構成

図1にASN.1処理系のソフトウェア構成を示す。コンパイラがASN.1定義のうち、静的定義部の符号化/復号関数とインタプリタのためのASN.1定義の内部表現を生成する。動的定義部はASN.1処理系が利用者プログラムの実行時に構文解析し、内部表現の追加、更新を行う。インタプリタがこの内部表現とコンパイラの生成した符号化/復号関数を用いて符号化/復号処理を行う。以降では、特にASN.1定義の内部表現、部分符号化/復号のためのASN.1情報の内部表現および応用プログラムインタフェースについて述べる。

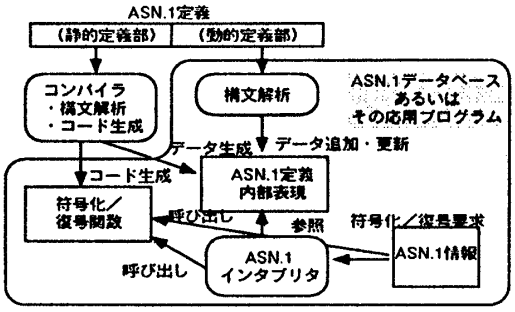

図1: ソフトウェア構成

4. ASN.1定義の内部表現

ASN.1定義の内部表現を図2に示す。

- 各々のASN.1型をASN.1型ノードと呼ぶ構造体で表現する。
- 動的定義部については、ASN.1型ノードは、型参照名、ASN.1型、付帯情報(ENUMERATED型の定数値リストやサブタイプによる最大データ長制限等)を持ち、さらに型を構成する要素の数と要素情報(タグ、付与されたラベル、OPTIONAL指定の有無、静的定義/動的定義を識別するフラグ、そのASN.1型ノードへのポインタ)のリストへのポインタからなる。(図2(a))
- 静的定義部については、ASN.1型ノードは型参照名、ASN.1型、生成された符号化関数および復号関数へのポインタを格納する。(図2(b))

5. ASN.1情報の内部表現

ASN.1定義によるデータ型を持つ情報(ASN.1情報)の内部表現を図3に示す。この内部表現には符号化されたオクテット列と復号された部分を表現するASN.1情報構造体が混在できる。コンパイラが生成する符号化/復号関数もインタプリタもこの内部表現を使用する。

符号化オクテット列内の復号済部分の表現　部分符号化/復号を可能とするためには、符号化されたオクテッ

"Design of High-Performance ASN.1 Encoder/Decoder for ASN.1 Database" by Satoshi NISHIYAMA, Hiroki HORIUCHI, Chihiro ONO, Sadao OBANA and Kenji SUZUKI, KDD R & D Laboratories

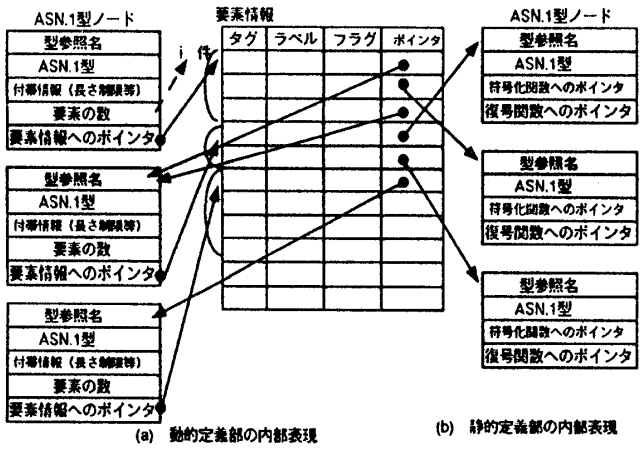

図 2: ASN.1 定義の内部表現

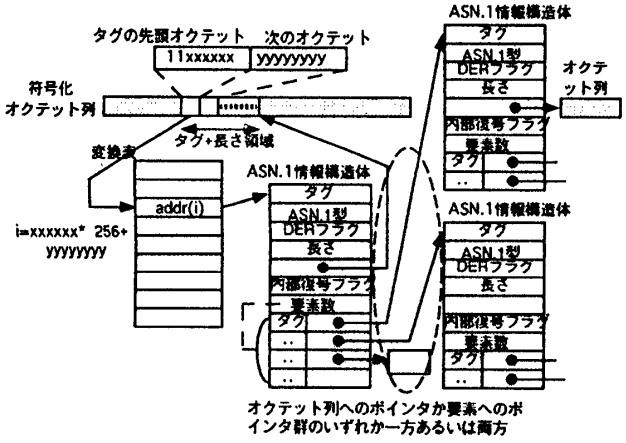

図 3: ASN.1 情報の内部表現

ト列の一部が既に復号されていることを、符号化規則として表現できる必要がある。ここでは、私的タグが殆んど用いられていないことを利用し、私的タグを用いて部分復号されていることを示す。具体的にはタグの先頭 1 オクテットのうち私的タグを表現するための先頭 2 ビットを除く 6 ビットと、それに続く 1 オクテット使用して ASN.1 情報構造体を識別する。

ASN.1 情報構造体の構造　復号部分を表現する ASN.1 情報構造体は ASN.1 型単位に ASN.1 情報を保持する。ASN.1 情報構造体は表 1に示す情報を持つ。

6. 応用プログラムインタフェース

- この ASN.1 処理系の利用者 (ASN.1 データベースやその応用プログラム) は、関数呼び出しにより符号化/復号処理をインタプリタに依頼する。また、利用者は静的定義部については直接 ASN.1 型毎の符号化/復号関数を呼び出すこともできる。
- 符号化時および復号時の引数と結果を図 4に示す。ASN.1 型を示す引数として各々の ASN.1 型に付与する内部識別子を使用する。この内部識別子は、静

表 1: ASN.1 情報構造体の持つ情報

フィールド			項目内容
タグ			符号化後のタグ値　(注 1)
ASN.1型			ASN.1型の種別(SET, INTEGER, Tagged等)を示す。
DERフラグ			既にDER符号化状態になっているか否かを示す。
符号化オクテット列の情報 (注2)	長さ		オクテット列の長さ
	ポインタ		オクテット列へのポインタ
	内部復号フラグ		オクテット列内でさらに部分復号されているか否かのフラグ
復号された情報 (注2)	要素数		その型に含まれる要素の数
	要素の情報	タグ	要素の符号化後のタグ値 (注1)．私的タグの値ならばオプショナルな値が省略されたことを示す
		ポインタ	要素のASN.1情報構造体へのポインタ、あるいは要素が基本形汎用型の場合はその情報を格納する領域へのポインタ

注 1) 構造形/基本形を示すフラグはASN.1定義からは決定できないため除く
注 2) 符号化オクテット列の情報と復号された情報の少なくとも一方が存在すること

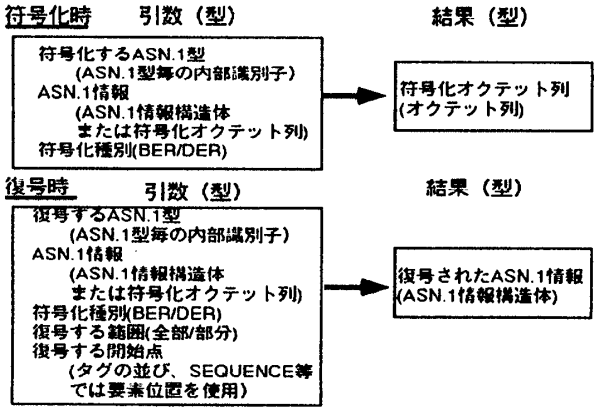

図 4: 符号化/復号時の引数と結果

的定義部については固定であるが、動的定義部については変化するため、型参照名から内部識別子への変換関数も提供する。復号時は一部のみの復号も可能とし、その場合復号の範囲とタグ (あるいは SEQUENCE 等では要素位置) の並びによる復号開始点を指定する。

7. おわりに

本稿では、ASN.1 データベースで重要となる高速な ASN.1 処理系の設計について報告した。コンパイラとインタプリタを組み合わせて高速性と動的な ASN.1 定義の変更機能を両立させ、そのための ASN.1 定義の内部表現と、部分符号化/復号のための ASN.1 情報の内部表現を示した。今後、この設計に基づき実装を進めていく予定である。最後に日頃御指導頂く KDD 研究所 浦野所長、真家次長に感謝します。

参考文献

[1] ITU-T 勧告 X.680 シリーズ,"ASN.1,"(1992).
[2] ITU-T 勧告 X.500 シリーズ, (1992).
[3] 西山 他: "ASN.1 データベースの実現方式に関する一考察," 第 49 回情処全大 4W-11, (1994).
[4] Hasegawa, T. et. al.: "Implementation and Evaluation of ASN.1 Compiler," J. of Info. Proc., Vol.15, No.2, (1992).
[5] 中川路 他: "OSI 抽象構文記法支援ソフトウェア APRICOT の開発と評価," 信学論文誌 D-I, Vol.J73-D-I, (1990).