

UMA FERRAMENTA PARA INTEGRAÇÃO ASN.1/ESTELLE

Maria Marta Leite

Laboratório de Integração Software/Hardware - LISHA
Departamento de Ciências Estatísticas e da Computação
Universidade Federal de Santa Catarina
88.049 - Florianópolis - SC
Telefone: (0482) 319739/319516
Fax: (0482) 341524
e-mail: cecimml@brufsc.bitnet

Jean-Marie Farines

Laboratório de Controle e Microinformática - LCMI
Departamento de Engenharia Elétrica
Universidade Federal de Santa Catarina
88.049 - Florianópolis - SC
Telefone: (0482) 319202
Fax: (0482) 341524
e-mail: lcmijmf@brufsc.bitnet

SUMÁRIO

Este artigo descreve uma ferramenta para integração da notação ASN.1 com a técnica de descrição formal Estelle. Dentre as possíveis abordagens para integração foi adotada a abordagem de tradução. A ferramenta se constitui de um tradutor de especificações de tipos e valores definidos em ASN.1 para especificações correspondentes em Estelle. Este tradutor está integrado em um ambiente de trabalho Estelle utilizado para tarefas de especificação e validação de protocolos.

ABSTRACT

This paper presents a tool which integrates ASN.1 notation with the specification language Estelle. Among all the possible approaches, we choose translation which allows to transform a specification written in ASN.1, in an equivalent Estelle specification. The resulting tool is integrated in a Estelle Environment which is used to specify and validate protocols.

1. INTRODUÇÃO

As especificações de protocolos e serviços fornecidas pelas normas ISO são geralmente descritas em linguagem natural, com máquinas de estado associadas, e as PDUs ("Protocol Data Unit") apresentadas através da notação ASN.1 ("Abstract Syntax Notation One") [ISO 87a] [ISO 87b].

A notação ASN.1 [ISO 87a] [ISO 87b] visa compatibilizar e tratar as diversas formas de representação da informação em sistemas heterogêneos. Através dela é possível definir estruturas de dados complexas de forma compacta e abstrata. Esta notação é principalmente utilizada para representação de dados nas camadas de apresentação e aplicação do modelo de referência OSI.

Por outro lado, a utilização de linguagem natural para especificar serviços e protocolos pode levar a ambigüidades, não-completude e inconsistência destes, e conseqüentemente à implementações que apresentam comportamentos indesejados. O uso de Técnicas de Descrição Formal (TDFs) permite evitar estes problemas, além de facilitar a construção de ferramentas de concepção automatizadas. A ISO propôs e normalizou duas técnicas de descrição formal, denominadas Estelle ("Extended State Transition Language") e LOTOS ("Language of Temporal Ordering Specification").

Neste trabalho nos interessamos particularmente pela linguagem de especificação Estelle por ser de fácil uso e por dispor de ferramentas automatizadas para simulação, verificação e geração de código. Entretanto, a estrutura de dados da linguagem Estelle é praticamente a mesma da linguagem Pascal, não sendo muito adequada para a representação de PDUs.

Este artigo apresenta uma proposta de integração da notação ASN.1 com a linguagem Estelle. Nos itens 2 e 3 deste artigo são apresentadas, respectivamente, a técnica de descrição formal Estelle e a notação abstrata ASN.1. A seguir, no item 4, discute-se algumas das possíveis abordagens para integração ASN.1/Estelle, destacando-se a abordagem escolhida. No item 5 é apresentada a correspondência entre estruturas ASN.1 e estruturas Estelle que servirá de base para o tradutor desenvolvido. Enfim, no item 6, é apresentado o tradutor e o ambiente no qual ele se insere.

2. A LINGUAGEM DE ESPECIFICAÇÃO ESTELLE

2.1. Características Gerais

A linguagem de especificação Estelle [ISO 89] é baseada no modelo de Máquina de Estado Finito Estendida (MEFE) onde a comunicação é realizada através de filas FIFO (na sua versão normalizada). As estruturas de dados são oriundas da linguagem Pascal, com algumas extensões e restrições.

Uma especificação Estelle descreve uma arquitetura hierarquizada de módulos cujas instâncias estão interconectadas entre si.

Os canais ("channel") definem os papéis ("roles") assumidos pelo módulo e o conjunto de interações (ou mensagens) que podem ser trocadas entre dois módulos. Estas mensagens são manipuladas através do ponto de interação ("ip") ao qual está associada uma fila FIFO não limitada. Estas filas podem ser exclusivas a um ponto de interação ("individual queue") ou compartilhadas por vários pontos ("common queue").

Um módulo consiste de dois componentes principais: o cabeçalho e o corpo. O cabeçalho ("header") define a visibilidade externa indicando os pontos de interação e as variáveis exportadas. O corpo ("body") define a implementação do cabeçalho, sendo que vários corpos podem ser associados ao mesmo cabeçalho, ou seja, módulos podem ter a mesma visibilidade externa e diferentes implementações. O corpo de um módulo divide-se em três partes: declaração, inicialização e transição.

A parte declaração é utilizada para declarar constantes, variáveis, tipos, estados de controle da MEFE, funções e procedimentos. A parte inicialização pode conter: inicialização da variável de estado, inicialização das variáveis adicionais e estabelecimento de ligações entre pontos de interação. As ligações entre pontos de interação são feitas através das cláusulas "attach" e "connect". A parte transição descreve o comportamento do módulo através de um conjunto de transições que definem o funcionamento da MEFE. Cada transição é constituída de duas partes: condições e ações. As condições são expressas através de cláusulas próprias de Estelle tais como: "priority", "when", "provided", "from" e "delay". As ações são expressas por uma seqüência de instruções Pascal e extensões Estelle.

2.2. A Representação dos Dados

Os tipos de dados Estelle, derivados do Pascal, incluem os tipos simples, os tipos estruturados e os tipos ponteiro. Aos tipos simples foi acrescentada a construção "...", que possibilita a declaração de um identificador sem atribuição de tipo específico ao mesmo.

O uso da linguagem Estelle para a especificação formal de protocolos tem mostrado as limitações desta técnica no que se refere ao poder de expressão das estruturas de dados. Isso leva a alguns problemas oriundos do Pascal [Courtiat 87]:

- imposição de uma definição explícita de todos os tipos utilizados;
- imposição de declarações estáticas para elementos essencialmente dinâmicos, tais como pontos de interação

Existem problemas na representação de PDUs em Estelle principalmente em virtude da complexidade das PDUs dos protocolos de mais alto nível do modelo OSI. A forma de representação dos dados utilizada por Estelle gera definições extensas e que algumas vezes levam em si escolhas de implantação. Notações mais abstratas (tal como ASN.1) permitem evitar estes problemas.

3. A NOTAÇÃO PARA SINTAXE ABSTRATA ASN.1

Dentro do contexto do modelo OSI, dois sistemas de comunicação devem poder interpretar da mesma maneira a estrutura e o conteúdo dos dados trocados pelas aplicações que eles suportam. Em consequência, deve existir um formato comum para os dados que permita sua transferência em meio heterogêneo. Este formato comum, chamado Sintaxe de Transferência, é negociado pelas entidades de Apresentação que asseguram as funções de codificação/decodificação entre as representações internas locais de dados (Sintaxe Fonte e Sintaxe Destino) e a Sintaxe de Transferência. Estas sintaxes (fonte, destino e de transferência) são três representações concretas diferentes para os mesmos dados. A sua estrutura pode ser descrita através da notação de sintaxe abstrata ASN.1, de forma independente da implementação. A figura 3.1 mostra os formatos de transferência de dados entre duas entidades de aplicação.

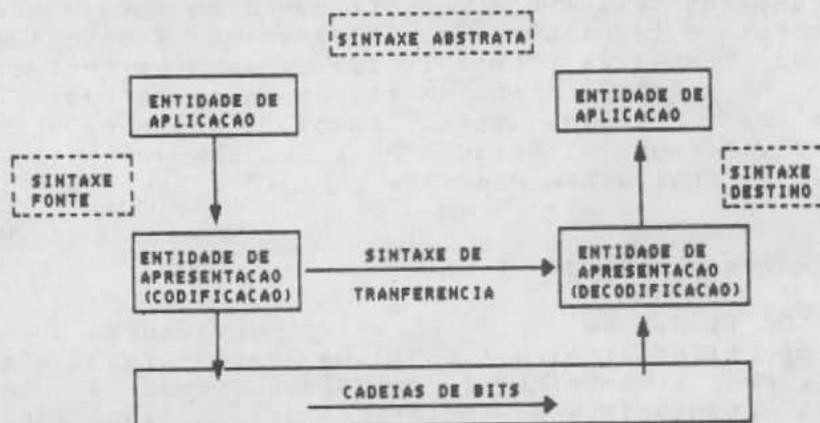


Figura 3.1

A norma ASN.1 contém a definição da notação ASN.1 e a definição das regras de codificação/decodificação aplicáveis a valores ASN.1.

A definição de PDU de uma aplicação é contida em um módulo. No interior do módulo, a notação permite que sejam definidos tipos sob a forma de árvore, adaptada à definição de uma sintaxe abstrata.

As tabelas 3.1 e 3.2 mostram, respectivamente, os principais tipos primitivos e os tipos construtores de ASN.1.

INTEGER	INTEIRO DE TAMANHO QUALQUER
BOOLEAN	'TRUE' OU 'FALSE'
BIT STRING	LISTA EVENTUALMENTE NULA DE BITS
OCTET STRING	LISTA EVENTUALMENTE NULA DE OCTETOS
ANY	UNIAO DE TODOS OS TIPOS ASN.1
NULL	AUSENCIA EFETIVA DE ELEMENTO
OBJECT IDENTIFIER	NOME DE OBJETO (EX: NOME DE MODULO ASN.1)

Tabela 3.1: Tipos Primitivos

SEQUENCE	LISTA ORDENADA DE ELEMENTOS DE TIPOS DIFERENTES
SEQUENCE OF	LISTA ORDENADA DE ELEMENTOS DE MESMO TIPO
SET	LISTA NAO ORDENADA DE ELEMENTOS DE TIPOS DIFERENTES
SET OF	LISTA NAO ORDENADA DE ELEMENTOS DE MESMO TIPO
CHOICE	UM TIPO ESCOLHIDO ENTRE UMA LISTA DE TIPOS

Tabela 3.2: Tipos Construtores

A notação ASN.1 possui outros tipos auxiliares que definem elementos opcionais (OPTIONAL) ou valores pré-determinados (DEFAULT) numa seqüência.

ASN.1 tem ainda a possibilidade de utilizar etiquetas (TAGs) que servem para identificar tipos ou campos e é composta por uma classe e um número. São definidas quatro classes de etiquetas: UNIVERSAL, para os tipos ASN.1 normalizados; APPLICATION, para os tipos ASN.1 definidos pelos padrões específicos de protocolos; PRIVATE, para tipos ASN.1 não definidos nos documentos ISO e "Context Specific", como "default" quando nenhuma das três classes for utilizada.

A utilização de diferentes tipos da notação ASN.1 é ilustrado na figura 3.2 [ISO 87a].

```

PersonnelRecord ::= [APPLICATION 0] IMPLICIT SET
(
  name      Name,
  title     [0] VisibleString,
  number    EmployeeNumber,
  dateOfHire [1] Date,
  nameOfSpouse [2] Name,
  children  Children [3] IMPLICIT SEQUENCE OF
              ChildInformation DEFAULT ( )
)

ChildInformation ::= SET
(
  name      Name,
  dateOfBirth [0] Date )
Name ::= [APPLICATION 1] IMPLICIT SEQUENCE
(
  givenName VisibleString,
  initial   VisibleString,
  familyName VisibleString )
EmployeeNumber ::= [APPLICATION 1] IMPLICIT INTEGER
Date ::= [APPLICATION 3] IMPLICIT VisibleString -- YYYYMMDD
  
```

Figura 3.2

4. ABORDAGENS PARA INTEGRAÇÃO ASN.1 - ESTELLE

A integração da notação ASN.1 com a linguagem Estelle vem sendo feita principalmente segundo as seguintes abordagens [Bochmann 90]:

- **SUBSTITUIÇÃO:** nesta abordagem a notação ASN.1 é utilizada para substituir os tipos Pascal para representação de dados nas especificações Estelle. é importante notar que ASN.1 não permite acessar elementos individuais em uma estrutura de dados. Entretanto, este acesso se faz necessário para escrever as partes da especificação relacionadas a interpretação e seleção de valores de parâmetros. Neste caso é necessário utilizar uma notação ASN.1 com extensões que permita o acesso a parâmetros, ao invés das formas de acesso proporcionadas por Estelle. Esta abordagem foi utilizada em [Alvey 87] e [Costa 89].

- **COMBINAÇÃO:** nesta abordagem a situação é similar a anterior, exceto que as construções de Estelle permanecem válidas e a notação ASN.1 é utilizada como um método de descrição alternativa de tipos e valores dentro da TDF. Isto leva a uma duplicação de funções na linguagem de especificação. Outro problema está relacionado com o impacto que as mudanças na linguagem podem provocar no lado de descrição do comportamento do protocolo, uma vez que são necessárias modificações na definição de Estelle que permitam acessar as estruturas ASN.1 [Hasegawa 88].

- **TRADUÇÃO:** nesta abordagem construções ASN.1 são traduzidas para construções equivalentes na TDF. As construções resultantes da tradução devem ser integradas a uma especificação formal completa do protocolo descrito em Estelle. Nesta abordagem não há necessidade de extensões à ASN.1 nem modificações na definição da linguagem Estelle. Por conseguinte, continuam utilizáveis as ferramentas existentes para Estelle e ASN.1 [Bochmann 90].

Nota-se que tanto a substituição quanto a combinação exigem alterações em Estelle ou extensões à ASN.1. Na abordagem de tradução estas permanecem inalteradas. Entretanto, para que esta abordagem seja conveniente é necessário que as definições de PDU resultantes da tradução sejam:

- simples e legíveis para o especificador de protocolo
- passíveis de integração numa especificação formal completa de um protocolo descrito em Estelle

Para o presente trabalho optou-se pela abordagem de tradução. A integração de ASN.1 com Estelle sem propor qualquer alteração na notação abstrata ou na linguagem de especificação e a possibilidade de utilização de ferramentas já construídas foram os fatores decisivos para na escolha.

5. UMA PROPOSTA DE TRADUÇÃO ASN.1/ESTELLE

5.1. Premissas para a Tradução

Algumas condições devem ser atendidas para que a tradução possa ser feita de forma automática e otimizada. Estas condições serão discutidas a seguir.

5.1.1. Os Tipos ASN.1

A tradução foi feita levando em consideração as similaridades existentes entre as declarações ASN.1 e as declarações Pascal que se encontram na linguagem Estelle.

Entretanto, o poder de expressão dos tipos ASN.1 é significativamente mais rico que os tipos da linguagem Estelle. Comparando as declarações ASN.1 com as de Estelle destacamos para as primeiras:

- tamanho para os tipos indefinido
- definição de campos OPTIONAL, sem similar em Estelle
- definição de valores DEFAULT, pois os componentes de um tipo Pascal não são definidos a não ser pelos seus tipos

É importante notar que a maioria das construções ASN.1 são traduzidas para Estelle de forma simples. Para os casos onde não há construções Pascal similares, a tradução utilizada levou em consideração a simplicidade na correspondência e a eficiência na implementação.

5.1.2. A Codificação de Valores

Existem informações incluídas em definições ASN.1 que são relacionadas com a codificação de valores, tais como:

- TAG's e seus valores numéricos;
- indicação IMPLICIT;
- distinção entre SEQUENCE e SET;
- distinção entre SEQUENCE OF e SET OF.

Estas informações não são requeridas para a descrição do comportamento lógico da entidade de protocolo e portanto não deverão ser representadas nas definições Estelle obtidas pela tradução. Tais informações podem ser incluídas sob forma de comentários para que a correspondência com a definição original em ASN.1 seja mantida.

O uso da palavra IMPLICIT indica que a etiqueta original do tipo deve ser substituído por uma nova etiqueta especificada, ou seja, a etiqueta original é perdida. O uso de IMPLICIT reduz o "overhead" de transmissão [Perpignan 89] e indica apenas a informação necessária para codificação.

5.1.3. O Uso de Etiquetas

A utilização de etiquetas (TAG) em ASN.1 tem por objetivo distinguir tipos de determinados campos numa instância de comunicação. Sua utilização permite identificar os campos através dos números de TAG somente, sem necessidade de identificadores. Os identificadores servem apenas para fins de documentação dos campos e podem ser utilizados no contexto de um tipo SET, SEQUENCE ou CHOICE. Tais identificadores, se utilizados, devem ser diferentes, de modo a não criar ambigüidades. Para que a tradução se torne mais eficiente, considerando a utilização de Estelle como linguagem alvo, a identificação de campos deve ser feita como no exemplo seguinte [Perpignan 89]:

```
Registro ::= CHOICE ( [0] NumericString    --CPF
                    [1] NumericString )   --Identidade
numRegistro Registro ::= "316120437"
```

A única forma possível para saber se o número corresponde ao CPF ou a Identidade consiste em nomear os campos, como segue:

```
Registro ::= CHOICE ( cpf      [0] NumericString
                    identidade [1] NumericString )
numRegistro Registro ::= cpf 316120437
```

Desta forma a declaração fica clara e passível de tradução embora as duas estejam sintaticamente corretas segundo a notação ASN.1.

5.2. Propostas de Tradução

5.2.1. Estruturas ASN.1 com Tradução Natural

INTEGER e BOOLEAN

Estes tipos pré-definidos tem representação similar em Estelle. Um inteiro em ASN.1 não tem limite de tamanho. Em Estelle também não é definido este limite, porém, a maioria dos compiladores Estelle impõe um tamanho máximo dos inteiros devido à restrições impostas pela linguagem utilizada para geração de código.

ANY

Este tipo ASN.1 corresponde a uma escolha entre os vários tipos definidos na notação; esta escolha será feita em função da implementação. Em Estelle este tipo pode ser representado pela construção "...", que significa que o tipo não será definido até que mais informações sejam fornecidas pela implementação.

SEQUENCE e SET

Os tipos SEQUENCE e SET de ASN.1 contém uma quantidade de elementos de tipos diferentes. A única diferença entre as duas construções é a ordem de transmissão destes elementos que é ordenada no caso de SEQUENCE e

indefinida para o caso de SET. Esta diferença será observada apenas na codificação, não tendo significado na definição da estrutura. Sendo assim, elas podem ser traduzidas para a mesma estrutura em Estelle, que é o record do Pascal. Por exemplo, a definição ASN.1:

```
Tipo1 ::= SEQUENCE ( b BOOLEAN, i INTEGER )
```

Deveria ser traduzida para a linguagem Estelle como:

```
Tipo1 = record                                     (*SEQUENCE*)
    b:BOOLEAN;
    i:INTEGER;
end;
```

CHOICE

O tipo CHOICE é utilizado para modelar uma variável que é selecionada a partir de uma coleção de variáveis cujo número é conhecido e pequeno. A tradução para Estelle é feita através da declaração record com variantes de Estelle. No exemplo seguinte Tipo2 pode ser inteiro ou booleano, e se escreve em ASN.1:

```
Tipo2 ::= CHOICE ( i INTEGER
                  b BOOLEAN )
```

Sua tradução será:

```
Tipo2 = record                                     (*CHOICE*)
    case choice_Tipo2: integer of
        Tipo2_i: ( i : INTEGER);
        Tipo2_b: ( b : BOOLEAN);
    end;
end;
```

5.2.2. Estruturas ASN.1 com Formas Especiais de Tradução

Enquanto as traduções apresentadas anteriormente são relativamente simples, as estruturas apresentadas a seguir não possuem correspondência imediata em Estelle e necessitam ser modeladas por estruturas de dados compostas.

OCTET STRING e BIT STRING

Em ASN.1 os tipos OCTET STRING e BIT STRING são utilizados para modelar dados binários onde a quantidade de elementos não é especificada ou será especificada em algum outro ponto. O tamanho em bits de cada elemento é um múltiplo de 8(oito) para OCTET STRING e qualquer para BIT STRING. Existem várias especializações para OCTET STRING diferindo entre si nos tipos de valores de octetos permitidos. Por exemplo, um IA5String permite a utilização dos caracteres padrão ASCII usual e um NumericString permite a utilização apenas dos dígitos numéricos.

Estelle não tem o tipo próprio definido para cadeias de caracteres. Propõem-se então utilizar a estrutura array por ser uma forma simples e existente em qualquer linguagem

alvo utilizada pelas ferramentas para Estelle. O array deve possuir definição de tamanho máximo e para que a tradução seja possível. Sendo assim, é necessário que este tamanho (tam_OCTETSTRING e tam_BITSTRING, no exemplo abaixo) seja definido pela especificação. Por exemplo as especificações ASN.1:

```
Tipo3 ::= OCTET STRING
Tipo4 ::= BIT STRING
```

podem ser traduzidas em Estelle como:

```
Tipo3 = array [1..tam_OCTETSTRING] of octeto;
Tipo4 = array [1..tam_BITSTRING] of binario;
```

onde:

```
octeto = 0..255
binario = 0..1
```

NULL

é utilizado nos tipos estruturados ASN.1 para indicar a possibilidade de ausência efetiva de algum elemento. Uma vez que a utilização do tipo NULL esta sempre relacionada com o tipo CHOICE, um tipo definido como NULL pode ser traduzido como um tipo indefinido associado em um "record" com variantes. Por exemplo temos:

```
Tipo5 ::= CHOICE ( b BOOLEAN
                  i INTEGER
                  NULL )
```

cuja tradução será:

```
Tipo5 = record                                (*CHOICE*)
  case choice_Tipo5 : INTEGER of
    tipo5_b      : ( b : BOOLEAN ) ;
    tipo5_i      : ( i : INTEGER ) ;
    tipo5-null   : ;
  end;
end;
```

OPTIONAL

Um elemento dentro de uma estrutura SEQUENCE ou SET pode ser declarado opcional. Neste caso a tradução consiste em adicionar um componente booleano imediatamente anterior (ou posterior) ao campo definido como OPTIONAL indicando se o campo existe ou não. Por exemplo temos a definição de tipo ASN.1:

```
Tipo6 ::= SEQUENCE ( i INTEGER OPTIONAL,
                    b BOOLEAN )
```

cuja tradução será:

```
Tipo6 = record                                (*SEQUENCE*)
  i_presente: BOOLEAN;
  i: INTEGER;
  b: BOOLEAN;
end;
```

SEQUENCE OF e SET OF

A tradução destas estruturas é a que apresenta mais dificuldade em virtude da não existência de um tipo correspondente em Estelle. Um valor do tipo SEQUENCE OF <tipo> é uma sequência ilimitada de valores, cada um do tipo especificado. Um valor do tipo SET OF <tipo> tem definição similar, exceto que a ordem dos elementos não é significativa. A mesma forma de tradução para Estelle será utilizada por não existir diferença fundamental entre elas. Neste caso a tradução será feita utilizando uma estrutura de lista com ponteiros. Por exemplo:

```
Tipo7 ::= SEQUENCE OF INTEGER
```

será traduzido para:

```
Tipo7 = !Lista_Tipo7;          (*SEQUENCE OF*)
Lista_Tipo7 = record
    item: integer;
    próximo: !Lista_Tipo7
end;
```

DEFAULT

Esta opção deve ser considerada apenas na fase de inicialização de variáveis uma vez que Estelle não permite a inicialização de variáveis juntamente com sua declaração.

A tabela 5.1 apresenta um resumo das formas de tradução adotadas para a ferramenta de tradução ASN.1/Estelle.

ASN.1	ESTELLE
INTEGER	integer
BOOLEAN	boolean
BIT STRING	array [1..tam_BITSTRING] of binario (onde binario = 0..1)
OCTET STRING	array [1..tam_OCTETSTRING] of octeto (onde octeto = 0..255)
ANY	...
NULL	record (com variantes)
SEQUENCE / SET	record
SEQUENCE OF / SET OF	lista com ponteiros
CHOICE	record (com variantes)

Tabela 5.1: Resumo das Traduções

6. O TRADUTOR ASN.1/ESTELLE E SUA INSERÇÃO NUM AMBIENTE GERAL

Existem atualmente uma série de ambientes para Estelle tais como ESTIM [Saqi 88], VEDA [Jard 88], XESAR [Richier 87], VALIRA [Vuong 86] e GROPE [New 89] entre

outros. A ferramenta de tradução que foi desenvolvida pode ser agregada a um ambiente de simulação e geração de código para especificações Estelle conforme a figura 6.1.

A estrutura do tradutor é composta por quatro módulos que realizam as seguintes funções:

- análise léxica
- análise sintática
- verificações semânticas
- geração de estruturas Estelle

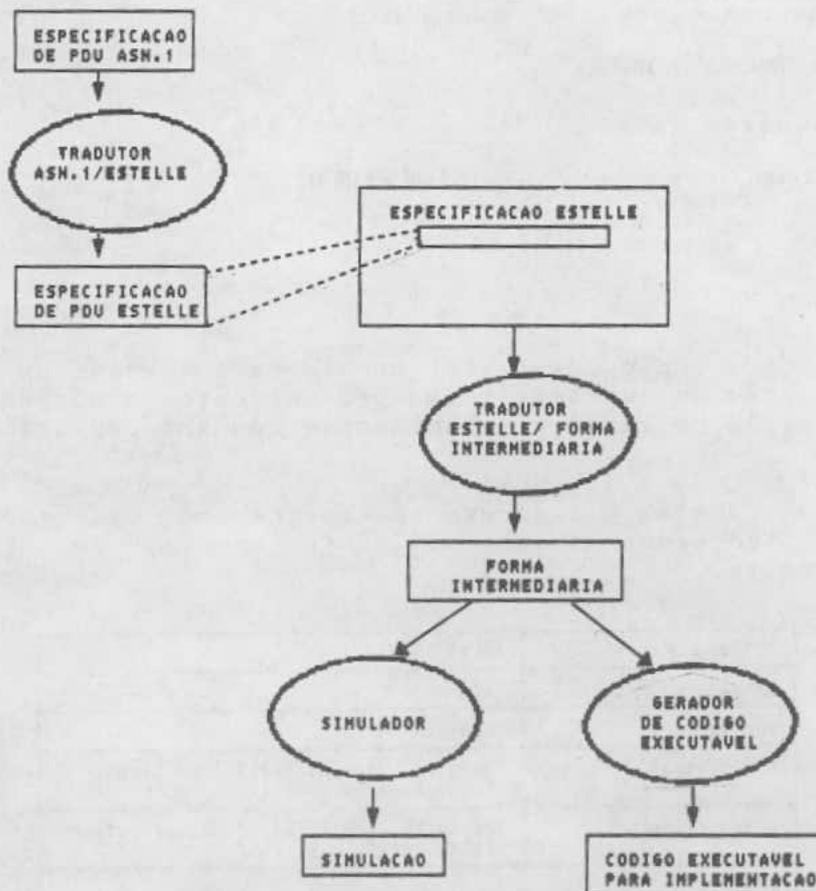


Figura 6.1

O tradutor foi desenvolvido utilizando as ferramentas LEX [Lesk 75], para a fase de análise léxica, e YACC [Johnson 75] para as fases de análise sintática, verificações semânticas e geração de código. Ele encontra-se atualmente em fase de testes.

Como ilustração da utilização do tradutor, apresentamos na figura 6.2 a tradução do exemplo utilizado na figura 3.2.

As dificuldades encontradas na implementação dizem respeito principalmente à problemas de sensibilidade ao contexto e ambigüidade em ASN.1. Por exemplo, a introdução de separadores (colchetes) na notação de valores dos tipos

CHOICE, SELECTION e ANY é necessária para evitar "look-ahead" infinito durante a fase de análise sintática. Outro exemplo é o tipo NULL que possui a mesma notação para tipo e valor. Os problemas encontrados foram parcialmente solucionados através de alterações propostas no documento [ISO 87c].

```

(* MODULO : EXEMPLO *)

type PersonnelRecord = (* [ APPLICATION 0 ] IMPLICIT *) record (* SET *)
    name : Name ;
    title : (* [ 0 ] *) VisibleString ;
    number : EmployeeNumber ;
    dateOfHire : (* [ -1 ] *) Date ;
    nameOfSpouse : (* [ 2 ] *) Name ;
    children : (* [ 3 ] IMPLICIT *) ^ChildInformation_list;
end ;

type ChildInformation_list = record (* SEQUENCE OF *)
    item: ChildInformation;
    next: ^ChildInformation_list;
end;

type ChildInformation = record (* SET *)
    name : Name ;
    dateOfBirth : (* [ 0 ] *) Date ;
end ;

type Name = (* [ APPLICATION 1 ] IMPLICIT *) record (* SEQUENCE *)
    givenName : VisibleString ;
    initial : VisibleString ;
    familyName : VisibleString ;
end ;

type EmployeeNumber = (* [ APPLICATION 1 ] IMPLICIT *) INTEGER ;

type Date = (* [ APPLICATION 3 ] IMPLICIT *) VisibleString ;

```

Figura 6.2

As definições recursivas, macros e extensões recentes de ASN.1 não foram introduzidas nesta primeira versão da ferramenta de tradução.

7. CONCLUSÕES

Este artigo apresentou uma proposta de integração ASN.1/Estelle baseada na abordagem de tradução. A escolha desta abordagem baseou-se na simplicidade de sua utilização e implementação e no fato de não requerer qualquer modificação em Estelle ou em ASN.1. Uma ferramenta de tradução que implementa os princípios propostos foi também desenvolvida.

A tradução para a linguagem Estelle se justifica pela possibilidade de ter uma especificação formal completa e homogênea numa única linguagem de especificação. Uma ferramenta de tradução pode ser facilmente integrada num ambiente contendo outras ferramentas de concepção de sistemas de comunicação descritos em Estelle. Desta forma aumenta-se as facilidades de automatização da implementação de protocolos a partir de uma especificação formal.

Está ainda prevista a utilização da ferramenta de tradução em algumas aplicações e em particular no projeto de implantação do "Sistema Integrado Nacional para Pesquisas Bibliográficas e Apropriação de Informações para Pesquisadores" [KNOWBOT 91].

8. REFERÊNCIAS

- [Alvey 87] "Formal Methods Applied to Protocols (FORMAT): A Framework for the Underlying Concepts of Communications Protocols". Relatório no.4, Alvey Directorate, UK. 1987.
- [Bochmann 90] G.v.Bochmann, et alli. "Implementation Support Tools for OSI Application Layer Protocols". Technical Report no. 748. Université de Montréal. 1990.
- [Costa 89] Rosvelter J. C. da Costa. "Sur l'integration d'ASN.1 dans Estelle". DEA. Informatique, Université Paul Sabatier. Toulouse, França. 1989.
- [Courtat 87] J.P.Courtat. "Contribution à la Description Formelle de Protocoles". These de Doctorat d'Etat. Université Paul Sabatier, Toulouse. 1987.
- [Hasegawa 88] T.Hasegawa, et alli. "Automatic ADA Program Generation from Protocol Specifications based on Estelle and ASN.1". Proc. of International Conference on Computer Communications (ICCC), Israel. 1988.
- [ISO 87a] ISO 8824, Information Processing - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1). 1987.
- [ISO 87b] ISO 8825, Information Processing - Open Systems Interconnection - Basic Encoding Rules for Abstract Syntax Notation One (ASN.1). 1987.
- [ISO 87c] ISO/IEC JTC1/SC21/WG6 - Some ASN.1 Discussion Topics. Dezembro, 1987.
- [ISO 89] ISO IS 9074, "Estelle: A Formal Description Technique based on an Extended State Transition Model". 1989.

- [Jard 88] C. Jard, et alli. "Development of VEDA, a Prototyping Tool for Distributed Algorithms". IEEE Transactions on Software Engineering, vol.14, no.3, março, 1988.
- [Johnson 75] S.C.Johnson. "YACC - Yet Another Compiler-Compiler", Computer Science Technical Report no.32, Bell Laboratories. Julho, 1975.
- [KNOWBOT 91] PROTEM-CC-S, O Projeto KNOWBOT. Documento organizado por L.M.R.Tarouco. Agosto, 1991.
- [Lesk 75] M.E.Lesk. "LEX - A Lexical Analyser Generator". Computer Science Technical Report no.39, Bell Laboratories. Outubro, 1975.
- [New 89] S.T. Vuong, D.D. Hui, D.D. Cowan. "VALIRA - a tool for validation via reachability analysis". 6th IFIP Workshop on Protocol Specification, Testing and Verification. Montreal, junho, 1986.
- [Perpignan 89] D.M.Perpignan. "Tradutor Automático ASN.1 /X.409 - Aplicação em Sistemas de Tratamento de Mensagens". Anais do 7ª SBRC, Recife. 1989.
- [Richier 87] J-L. Richier et alli. "Verification in XESAR of the sliding window protocol". 7th IFIP Symposium on Protocol Specification, Testing and Verification, Zurich, Maio, 1987.
- [Saqui 88] P. de Saqui-Sannes, J-P. Courtiat. "ESTIM: The Estelle Simulator Prototype of the ESPRIT/SEDOS Project". Proceedings of the First International Conference on Formal Description Techniques, Stirling, setembro, 1988.
- [Vuong 86] D. New, P.D. Amer. "Adding Graphics and Animation to Estelle". Proceedings of the 9th International Symposium on Protocol Specification, Testing and Verification. Twente, The Netherlands, junho, 1989.