Réseaux et protocoles

Maitrise d'informatique ©

REPR
Td 12
99/00

Le niveau Présentation

But

Nous allons étudier le problème de la communication de données dans les réseaux hétérogènes où cohabitent diverses applications dont le format de représentation interne est différent.

1 Introduction

La niveau Présentation est la 6^e couche du modèle de l'ISO. Le fonctionnement des protocoles de Présentation est étroitement lié à ceux de ses niveaux adjacents. Sa spécificité réside en la mise à disposition du niveau Application de moyens de transcodage permettant la réalisation de réseaux hétérogènes où cohabitent diverses applications avec des formats de représentation internes est différents.

Question 1 *Donnez des exemples de représentation internes pouvant amener des problèmes dans l'échange des données.*

Obtenir des communications compréhensibles par tous les acteurs du niveau Application nécéssite de standardiser les échanges (à défaut de pouvoir standardiser les formats de représentation interne des applications). Il existe trois paramètres à considérer dans un échange de données :

1. Définition de ce qui va être échangé dans une *syntaxe abstraite*.
2. Représentation interne à chaque application de ce qui va être échangé.
3. Transfert interprétable par les correspondants de ce qui est échangé au moyen de *règles de codage*.

Clairement, le travail de normalisation ne s'applique qu'aux points 1 et 3, le point 2 restant à la charge de l'implémenteur.

Question 2 *Quelles étapes sont nécessaires dans la réalisation d'une application normalisée. Donnez le schéma de principe d'une communication.*

Le rôle de la notation de syntaxe abstraite ASN1, normalisée par l'ISO, est de permettre la définition d'un format commun de description des A-PDU échangées. Les règles de codage d'ASN1, quant à elles, s'appliquent au transfert des données échangées; l'ISO a normalisé trois formats de règles d'encodage : les règles de codage de base (BER), les règles de codage canonique et les règles de codage spécialisées.

2 Notation de syntaxe abstraite

Pouvoir décrire les données à échanger nécessite de pouvoir *construire*, à la manière des langages de programmation évolués (PASCAL, ADA, etc.), des types *constructeurs* à partir de types *primitifs*. Les types constructeurs désignent des types de données structurées constituées d'une ou plusieurs données typées. Les types primitifs désignent des types de données ne pouvant se décomposer.

On distingue en ASN, les types primitifs (ou prédéfinis) suivants (en autre) :

- **BOOLEAN**: Vrai ou faux.
- **INTEGER**: Entier de longueur arbitraire.
- **REAL**: Réel de précision arbitraire.
- **BIT STRING**: Liste de bits.
- **OCTET STRING**: Liste d'octets.
- **VISIBLE STRING**: Chaine de caractères.
- **PRINTABLE STRING**: Chaine de caractères imprimables.
- **NUMERIC STRING**: Chaine de carectères chiffres.
- **NULL**: Vide.
- **ENUMERATED**: Type énuméré.

Puis des types constructeurs qui sont les types d'objets construits :

- **SEQUENCE**: Suite ordonnée d'éléments de divers types.
- **SEQUENCE OF**: Suite ordonnée d'éléments de même type.
- **SET**: Idem **SEQUENCE** mais non ordonnée.
- **SET OF**: Idem **SEQUENCE OF** mais non ordonnée.

Et enfin les constructeurs de type suivants :

ANY désigne un type arbitraire.

CHOICE désigne un choix entre plusieurs types définis.

Question 3 *Pourquoi ANY et CHOICE ne sont il pas des types constructeurs ? Pourquoi ENUMERATED n'est-t'il pas un type constructeur ?*

On donne ci dessous un exemple de description de syntaxe abstraite pour un enregistrement d'une personne.

```
nom      ::= VISIBLE STRING
Prenoms  ::= SEQUENCE OF Nom
Personne ::= SEQUENCE {
         nom,
         Prenoms,
         age INTEGER,
         matricule INTEGER,
         marie BOOLEAN
}
```

Les types construits sont identifiés par une majuscule en première position, les identificateurs de valeur n'en comportent pas.

La norme prévoit l'utilisation du mot clef OPTIONNAL pour désigner un champ optionnel (par exemple OPTIONNAL BOOLEAN).

Question 4 *Rappelez la structure de la PDU CC de niveau Transport. Proposez une notation de syntaxe abstraite pour la décrire.*

Question 5 *Sachant que les identificateurs de valeur ne sont pas interprétés par le logiciel de Présentation, quels problèmes peuvent survenir avec l'utilisation du mot clef OPTIONNAL ? Quels mécanismes faut-il mettre en place pour y remédier ?*

Une étiquette de type sert à définir de façon non ambiguë un type par rapport à un autre type qui lui serait isomorphe. Outre les étiquettes spécifiques au contexte, on distingue :

UNIVERSAL : réservée pour coder les types de base (par exemple INTEGER est en fait l'étiquette [UNIVERSAL 2].

APPLICATION : réservée pour les structures standardisées liées à des classes d'applications qui sont normalisées.

PRIVATE : Pour des applications particulières, spécifiques et non normalisées.

La définition d'un type étiqueté produit le type étiqueté proprement dit qui sera repéré après codage par l'identificateur général de type étiqueté suivi par le numéro particulier qui est indiqué entre crochets dans la définition. Ce type étiqueté devra normalement ensuite être suivi par l'identificateur du type de base dont le type étiqueté est dérivé, puis par la valeur de l'élément correspondant.

Question 6 *Cette solution peut faire apparaître des redondances. Quelles sont-elles ?*

Pour éliminer les redondances, ASN fournit un moyen de les supprimer avec l'utilisation du mot clef IMPLICIT.

3 Règles de base de codage

BER utilise la notation fondamentale dite *Type, Longueur, Valeur*. Elle a été spécifiée pour implémenter directement ASN1.

Chaque valeur transmise contient trois champs.

- L'identificateur de type.
- Longueur en octets du champ de données (bit de poids fort à 1 et les bits 1 à 7 sont mis à zéro si la longueur est non définie).
- Le champ de données suivi éventuellement d'un marqueur de fin de données (cas où la longueur n'est pas définie) qui est constitué de deux octets à zéro.

Question 7 *Expliquez l'avantage qu'il pourrait y avoir à ne pas préciser la longueur mais plutôt à utiliser systématiquement le marqueur de fin.*

Question 8 *La norme précise que l'utilisation de la longueur indéfinie ne doit être appliquée qu'aux types constructeurs. Expliquez ce choix.*

Le champ identificateur se décompose de la manière suivante :

- bit 8 et 7 : le type de l'étiquette (00 UNIVERSAL, 01 APPLICATION, 10 "spécifique", 11 PRIVATE).
- bit 6 : 0 si type primitif, 1 si type constructeur.
- bit 5 à 1 : valeur de l'étiquette, 11111 si la valeur de l'étiquette vaut plus de 30, les octets suivants (bit de poids fort à un sauf le dernier) codant la valeur.

Les longueurs jusqu'à 128 octets sont directement codées sur un seul octet avec bit de poid fort à 0. Les longueurs plus grandes utilisent 7 bits utiles par octet. Le premier octet à son bit de poid fort à 1 et les sept bits suivants indiquent, en nombre d'octets, la longueur de la longueur. Les autres octets codent la longueur effective.

Le codage du champ de données est effectué de la manière suivante:

- Booléens un octet à 0 pour FALSE, autres valeurs pour TRUE.
- Entiers en complément à 2.
- Les types énumérés sont codés avec la valeur cardinale de l'objet désigné.
- Les chaines de bits sont émises avec un octet en entête précisant combien de bits inutiles comporte le dernier octet de la chaine.
- Les chaînes de caractères imprimables sont en ASCII.
- La valeur NULL est indiquée par une longueur égale à 0 (bit de poids fort à zéro compris).
- Les séquences et les ensembles sont codés de la même façon: les champs apparaissent dans l'ordre de leur spécification.

Soit les spécification de type suivantes :

```
Type1 ::= VISIBLE STRING\\
Type2 ::= [APPLICATION 5] IMPLICIT VISIBLE STRING\\
Type3 ::= [2] Type2\\
Type4 ::= [2] IMPLICIT Type2\\
Type5 ::= [APPLICATION 3] IMPLICIT Type3
```

Question 9 *Donnez l'encodage de la chaine «Martin» dans chacun des types ci-dessus.*

Question 10 *Quels sont les types redondants ? Quels sont les types constructeurs ?*

Question 11 *Pourquoi ne peut on utiliser IMPLICIT pour ANY et CHOICE ?*