Committed to connecting the world

#ICT4SDG

Introduction to ASN.1

MAIN CONCEPTS

ASN.1 is a formal notation used for describing data transmitted by telecommunications protocols, regardless of language implementation and physical representation of these data, whatever the application, whether complex or very simple.

> Abstract Syntax Notation number One
> is a standard that defines a formalism
> for the specification of abstract data types.

The notation provides a certain number of pre-defined basic types such as:

- integers (`INTEGER`),
- booleans (`BOOLEAN`),
- character strings (`IA5String`, `UniversalString`...),
- bit strings (`BIT STRING`),
- etc.,

and makes it possible to define constructed types such as:

- structures (`SEQUENCE`),
- lists (`SEQUENCE OF`),
- choice between types (`CHOICE`),
- etc.

Subtyping constraints can be also applied on any ASN.1 type in order to restrict its set of values.

Unlike many other syntaxes which claim to be extensible, ASN.1 offers extensibility which addresses the problem of, and provides support for, the interworking between previously deployed systems and newer, updated versions designed years apart.

ASN.1 sends information in any form (audio, video, data, etc.) anywhere it needs to be communicated digitally. ASN.1 only covers the structural aspects of information (there are no operators to handle the values once these are defined or to make calculations with). Therefore it is not a programming language.

ASN.1 definition can be contrasted to the concept in ABNF of "valid syntax", or in XSD of a "valid document", where the focus is entirely on what are valid encodings of data, without concern with any meaning that might be attached to such encodings. That is, without any of the necessary semantic linkages.

> One of the main reasons for the success of ASN.1 is that this notation is associated with several standardized encoding rules such as the BER (Basic Encoding Rules), or more recently the PER (Packed Encoding Rules), which prove useful for applications that undergo restrictions in terms of bandwidth. These encoding rules describe how the values defined in ASN.1 should be encoded for transmission (i.e., how they can be translated into the bytes 'over the wire' and reverse), regardless of machine, programming language, or how it is represented in an application program. ASN.1's encodings are more streamlined than many competing notations, enabling rapid and reliable transmission of extensible messages -- an advantage for wireless broadband. Because ASN.1 has been an international standard since 1984, its encoding rules are mature and have a long track record of reliability and interoperability.
>
> An ASN.1 definition can be readily mapped (by a pre-run-time processor) into a C or C++ or Java data structure that can be used by application code, and supported by run-time libraries providing encoding and decoding of representations in either an XML or a TLV format, or a very compact packed encoding format.
>
> Tools on almost all operating systems support ASN.1. ASN.1 also supports popular programming languages such as Java, C and C++, as well as older ones including COBOL. As an example of ASN.1's universality, there are tools that have been ported to over 150 different computing platforms.
>
> There are a lot of well-tested ASN.1 tools that have been used for a long time. Using such tools, there are less likely to be costly delays in bringing new products to market or, even worse, recalling products based on new code that hasn't been sufficiently tested for flaws.
>
> ASN.1 is widely used in industry sectors where efficient (low-bandwidth, low-transaction-cost) computer communications are needed, but is also being used in sectors where XML-encoded data is required (for example, transfer of biometric information).

CASE STUDY

Suppose a company owns several sales outlets linked to a central warehouse where stocks are maintained and deliveries start from. The company requires that its protocol have the following features:

- the orders are collected locally at the sales outlets ;
- they are transmitted to the warehouse, where the delivery procedure should be managed ;

- an account of the delivery must be sent back to the sales outlets for following through the client's order.

This protocol can be specified with the two following ASN.1 modules:
-- Module sample (sample:01/2014)
-- See also the README file
-- See also the index of all ASN.1 assignments needed in this Recommendation

```
Module-order DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

Order ::= SEQUENCE {header  Order-header,
                    items   SEQUENCE OF Order-line
}

Order-header ::= SEQUENCE {
  number   Order-number,
  date     Date,
  client   Client,
  payment  Payment-method
}

Order-number ::= NumericString(SIZE (12))

Date ::= NumericString(SIZE (8)) -- MMDDYYYY


Client ::= SEQUENCE {
  name      PrintableString(SIZE (1..20)),
  street    PrintableString(SIZE (1..50)) OPTIONAL,
  postcode  NumericString(SIZE (5)),
  town      PrintableString(SIZE (1..30)),
  country   PrintableString(SIZE (1..20)) DEFAULT default-country
}

default-country PrintableString ::= "France"

Payment-method ::= CHOICE {
  check        NumericString(SIZE (15)),
  credit-card  Credit-card,
  cash         NULL
}

Credit-card ::= SEQUENCE {
  type         Card-type,
  number       NumericString(SIZE (20)),
  expiry-date  NumericString(SIZE (6))-- MMYYYY --
}

Card-type ::= ENUMERATED {
  cb(0), visa(1), eurocard(2), diners(3), american-express(4)}

Order-line ::= SEQUENCE {
  item-code  Item-code,
  label      Label,
  quantity   Quantity,
  price      Cents
}

Item-code ::= NumericString(SIZE (7))
```

```
Label ::= PrintableString(SIZE (1..30))

Quantity ::= CHOICE {
  unites        INTEGER,
  millimetres   INTEGER,
  milligrammes  INTEGER
}

Cents ::= INTEGER

Delivery-report ::= SEQUENCE {
  order-code  Order-number,
  delivery    SEQUENCE OF Delivery-line
}

Delivery-line ::= SEQUENCE {item      Item-code,
                            quantity  Quantity
}

END



Protocol DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

IMPORTS
  Order, Delivery-report, Item-code, Quantity, Order-number
    FROM Module-order;

PDU ::= CHOICE {
  question
    CHOICE {question1  Order,
            question2  Item-code,
            question3  Order-number,
            ...},
  answer
    CHOICE {answer1  Delivery-report,
            answer2  Quantity,
            answer3  Delivery-report,
            ...}
}

END
```

APPLICATIONS FIELDS

Originally standardized to specify data protocols in an open system interconnection (OSI) environment, ASN.1 has imposed its intrinsic advantages in many other popular fields.

A lot of application fields of ASN.1 are presented in Chapter 7 of the book "ASN.1. Communication between Heterogeneous Systems", byOlivier Dubuisson (© 1999).

Other application fields are also listed here. ASN.1 is a critical part of our daily lives; it's everywhere, but it works so well it's invisible.

STATE OF STANDARDIZATION

ASN.1 was first standardized in 1984 by the CCITT (International Telegraph and Telephone Consultative Committee, now called ITU-T,International Telecommunication Union - Telecommunication Standardization Sector) under the name "X.409 Recommendation". A little later, ISO (International Organization for Standardization) chose to adopt this notation and split this recommendation into two separate documents: the abstract syntax (ASN.1) and the encoding rules (BER). In 1985, the CCITT decided to collaborate with ISO on these two documents.

In 1987, ISO published these documents as 8824 and 8825 (only three new types of character strings are added). In 1988, ISO merged with the IEC (International Electrotechnical Commission) forming a joint technical committee called ISO/IEC JTC 1, which is now in charge of the ASN.1 standard.

For its Blue Book, in 1989, the CCITT published the X.208 and X.209 recommendations: a new release for the ASN.1 standard, which was provided with extensions resulting from a common work with the JTC 1.

For the last version (available since end of 2008), the ISO 8824 standard was split into four parts:

- ISO 8824-1 | ITU-T X.680: Specification of basic notation,
- ISO 8824-2 | ITU-T X.681: Information object specification,
- ISO 8824-3 | ITU-T X.682: Constraint specification,
- ISO 8824-4 | ITU-T X.683: Parameterization of ASN.1 specifications.

As far as encoding rules are concerned, ISO 8825 standard was split into seven parts:

- ISO 8825-1 | ITU-T X.690: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
- ISO 8825-2 | ITU-T X.691: Specification of Packed Encoding Rules (PER)
- ISO 8825-3 | ITU-T X.692: ASN.1 encoding rules: Specification of Encoding Control Notation (ECN)
- ISO 8825-4 | ITU-T X.693: ASN.1 encoding rules: XML Encoding Rules (XER)
- ISO 8825-5 | ITU-T X.694: ASN.1 encoding rules: Mapping W3C XML schema definitions into ASN.1
- ISO 8825-6 | ITU-T X.695: ASN.1 encoding rules: Registration and application of PER encoding instructions
- ISO 8825-7 | ITU-T X.696: ASN.1 encoding rules: Specification of Octet Encoding Rules (OER)

ASN.1:2002 AND LATER EDITIONS ARE, FROM NOW ON, STRONGLY RECOMMENDED. THE 1990-RELEASE IS NO LONGER AVAILABLE.

Presented during the January 1999 ASN.1 meeting, the encoding control notation (ECN) allows specifiers to define their own encoding rules by referencing standardized encoding rules and modifying some of their characteristics, or even to set up completely new ones.

ASN.1 has a long record of accomplishment, having been in use since 1984. It has evolved over time to meet industry needs, such as PER support for the bandwidth-constrained wireless industry and XML support for easy use of common Web browsers.

- References
- ASN.1

"fathers"
- More ...

ASN.1 reference card, by OSS Nokalva
[pdf - 2 p]

ASN.1: A Powerful Schema Notation for XML and Fast Web Services (by O. Dubuisson)
Slides: [pdf]

John Larmouth
Douglas Steedman
James E. White

Changing from ASN.1:1988 to ASN.1:2002

What is ASN.1 ?

Back to top