11장. 상위계층 (2) - 표현 계층

컴퓨터 네트워크

이번 시간의 학습 목표

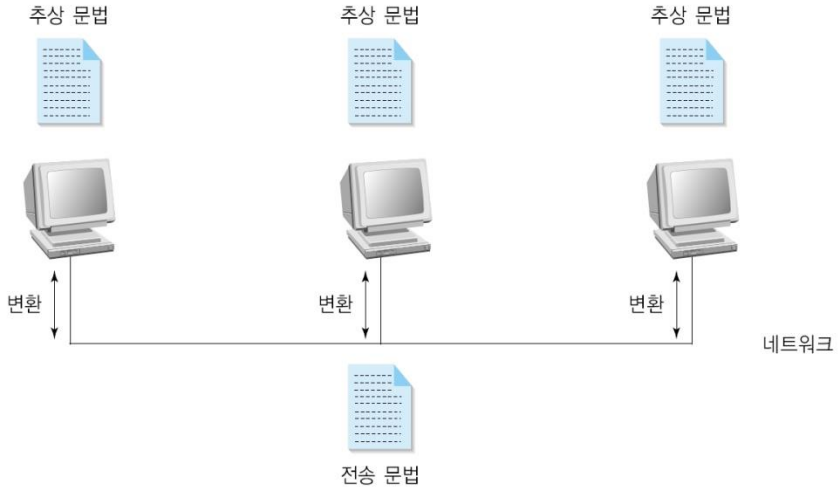
- 네트워크에서 데이터 표현의 일관성이 무엇인지 이해
- 데이터 압축 원리 이해

표현계층

- 응용 프로세스 사이에 전송되는 메시지의 표현방법과 관련
- 전송 메시지에 표현된 문법(syntax) 내용을 통신 양단의 프로세스가 해석하는 기능 제공
  - 송신 프로세스가 전달하려는 의미(Semantic)를 수신 프로세스가 정확히 이해하도록 함

데이터 표현

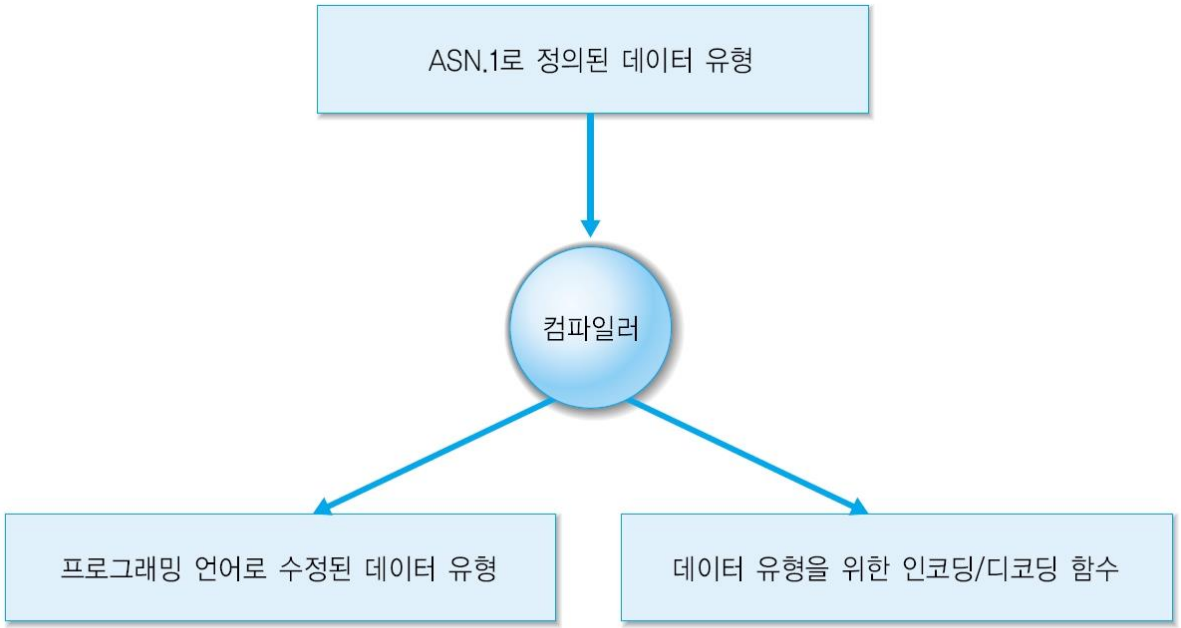
- 추상 문법과 전송 문법
  - 추상 문법: 각 컴퓨터에서 사용하는 데이터 표현 규칙
  - 전송 문법: 네트워크 전체에서 일관성을 지니는 표현 규칙
- 특정 컴퓨터에 독립적이면서 네트워크 전체에서 일관성을 가지는 새로운 표현 규칙(전송문법)으로 변환하여 전송
- 수신 측에서는 자신의 컴퓨터에서 이해하는 추상문법 형태로 변환하는 과정 필요



[그림 11-5] 추상 문법과 전송 문법

ASN.1 (1)

- 분산 환경에서 표현되는 데이터를 정의하기 위한 일반적인 추상 문법
- 다양한 언어용 코드를 생성하는 컴파일러 존재
- 응용계층에서의 문법규칙 뿐만 아니라 PDU 구조도 정의



[그림 11-6] ASN.1 컴파일러

ASN.1 (2)

- 지원하는 클래스 유형
  - UNIVERSAL: 일반 데이터 유형
  - CONTEXT-SPECIFIC: 특정 컨텍스트와 관련된 유형
  - APPLICATION: 응용 개체의 공통 유형
  - PRIVATE: 사용자 정의 유형
- UNIVERSAL 클래스
  - 일반 데이터 유형을 정의
  - 기본 유형(Primitive)과 구조(Constructed) 유형으로 구분
  - 기본 유형
    - 단위유형
    - 문자열 유형
    - BOOLEAN, INTEGER, BITSTRING, OCTSTRING, ……
  - 예

```
StudentNumber ::= INTEGER
Name ::= IA5String
MaleFemail ::= BOOLEAN
EntranceRight ::= BITSTRING {accept(0), reject(1)}
PersonalInformation ::= OCTETSTRING
```

ASN.1 (3)

- UNIVERSAL 클래스 (계속)
  - 구조 유형
    - UNIVERSAL SEQUENCE: 순서대로 나열한 임의 유형의 모임 (고정 크기)
    - SEQUENCE: 순서대로 나열한 동일 유형의 모임 (고정/가변 크기)
    - SET: 순서 없이 나열한 임의 유형의 모임 (고정 크기)
    - SETOF: 순서 없이 나열한 동일 유형의 모임 (고정/가변 크기)
    - CHOICE: 미리 정의된 유형 모임에서 선택한 순서가 없는 유형 모임 (고정 크기)

```
Student Record ::= SEQUENCE {
        StudentNumber   INTEGER,
        Name  IA5String,
        MaleFemail  BOOLEAN
}
```

ASN.1 (3)

- 태그
  - 구조 유형에 선언된 변수를 개별적으로 사용하기 위함
  - 배열의 인덱스와 유사
  - 선언 방법
    - CONTEXT-SPECIFIC: 태그의 범위가 현재 구조 유형에 한정됨
    - APPLICATION: 태그의 범위가 전체 응용 컨텍스트에 적용됨
    - PRIVATE: 태그의 범위가 해당 사용자에게 적용됨
  - 예

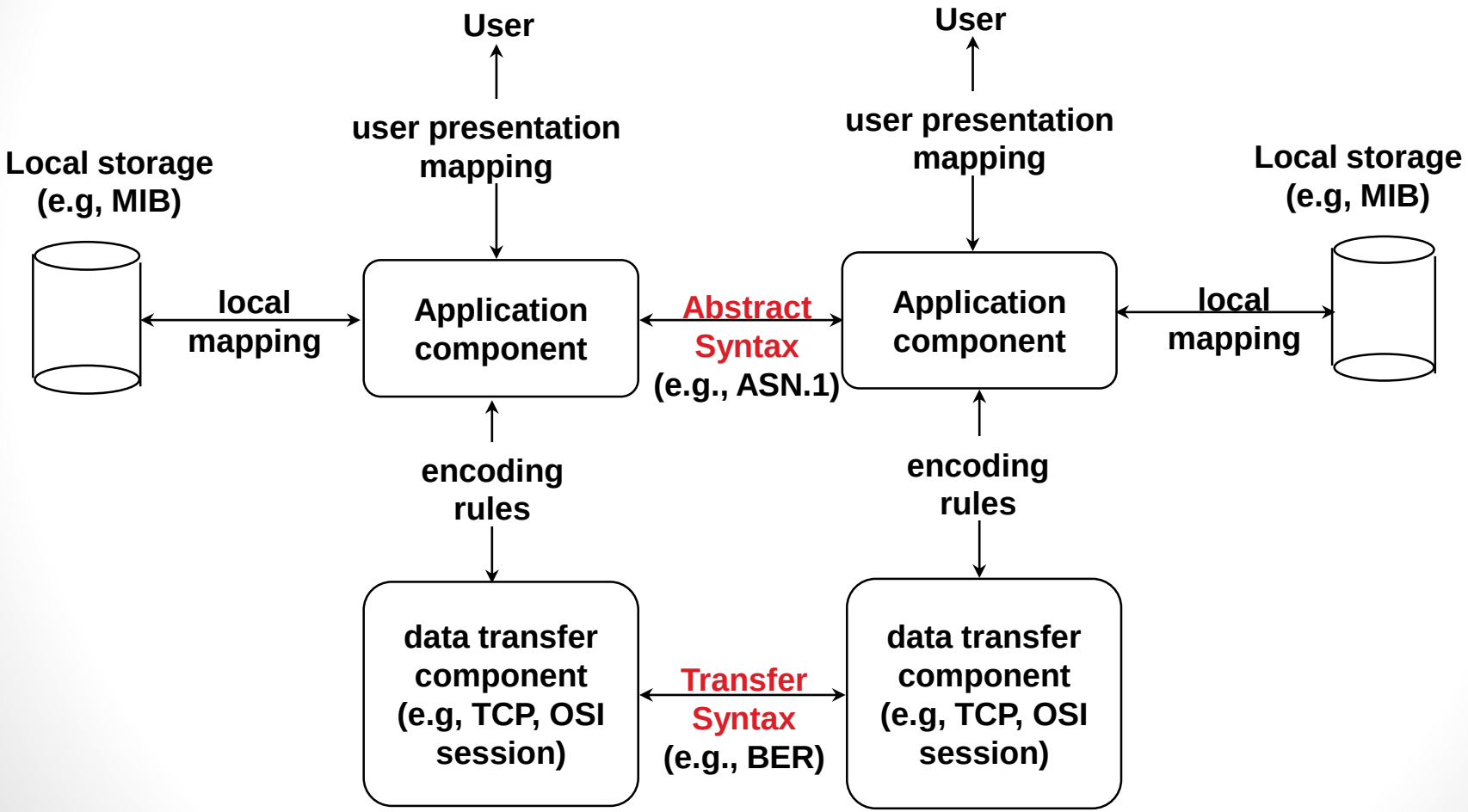
```
Student Record ::= SEQUENCE {
        StudentNumber [APPLICATION 1] INTEGER,
        Name [1] IA5String,
        MaleFemail [2]  BOOLEAN
}
```

    - StudentNumber는 다른 응용에서도 사용 가능
    - Name, MaleFemale은 현재 유형에서만 사용 가능

ASN.1 소개 (1)

- a machine independent data description language
- CCITT (X.208) and ISO (ISO 8824) standard
- define abstract syntax of application data
- define the structure of application and presentation protocol data units (PDUs)
- define SNMP, VOIP and IEC61850 MMS

ASN.1 소개 (2)


User
User
user presentation mapping
user presentation mapping
Local storage (e.g, MIB)
Local storage (e.g, MIB)
local mapping
Application component
Abstract Syntax (e.g., ASN.1)
Application component
local mapping
encoding rules
encoding rules
data transfer component (e.g, TCP, OSI session)
Transfer Syntax (e.g., BER)
data transfer component (e.g, TCP, OSI session)

ASN.1 소개 (3)

- Application Entity에 대한 규격 표현에 주로 사용
  - Application Entity
    - 정보의 의미(sematics)에만 의미를 두고 정보의 표현 형식(syntax)에는 무관 함
- 추상구문을 기술하기 위한 기법
  - Application/Presentation Layer PDU(Protocol Data Unit) 표현
- 다양한 인터페이스나 통신매체를 통하여 전송되어야 할 정보를 기술하기 위한 INL(Interface Notation Language)로 사용

ASN.1 소개 (4)

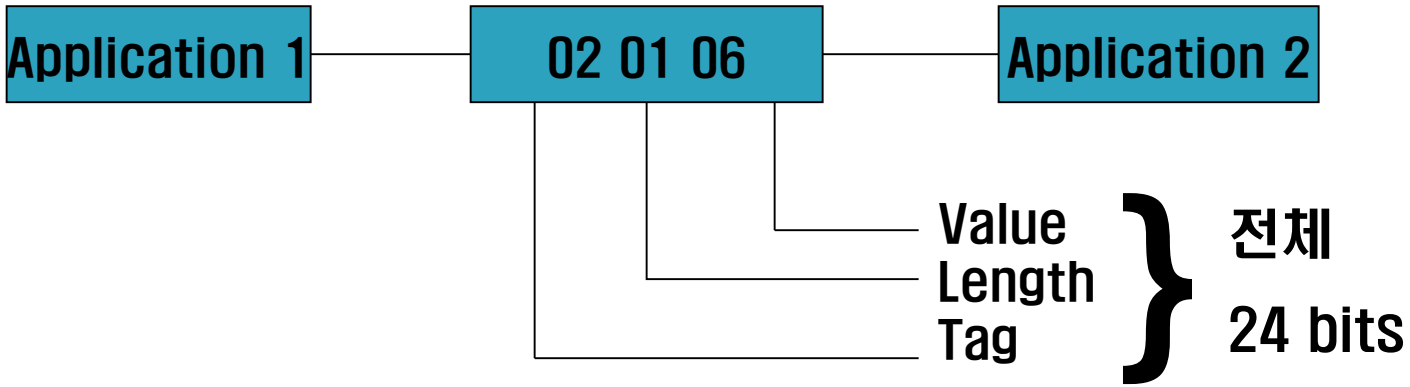
- 다양한 데이터 타입들의 표현

BOOLEAN	SEQUENCE
INTEGER	SEQUENCE OF
BIT STRING	GeneralizedTime
OCTET STRING	IA5String
NULL	BMPString
REAL	UTF8String
ENUMERATED	

BER(Basic Encoding Rules)

- BER 은 전송되는 데이터가 어떻게 인코딩 되어야 하는지를 기술
- 기기특성, 프로그래밍 언어, 에플리케이션 표현방법과 독립적

Age ::= INTEGER (0..7)
firstGrade Age ::= 6

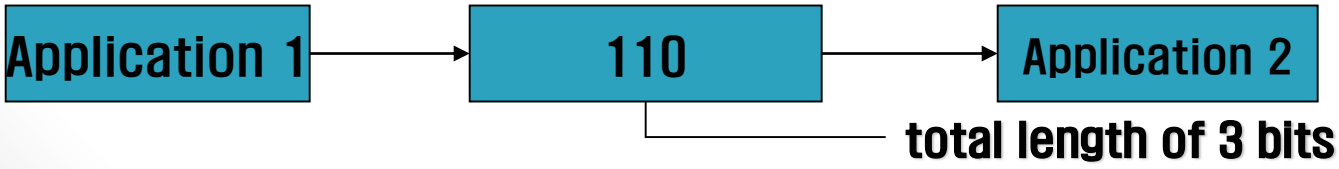


- CCITT (X.209) and ISO (ISO 8825) standard
- based on the use of a type-length-value (TLV) structure

PER-Packed Encoding Rules

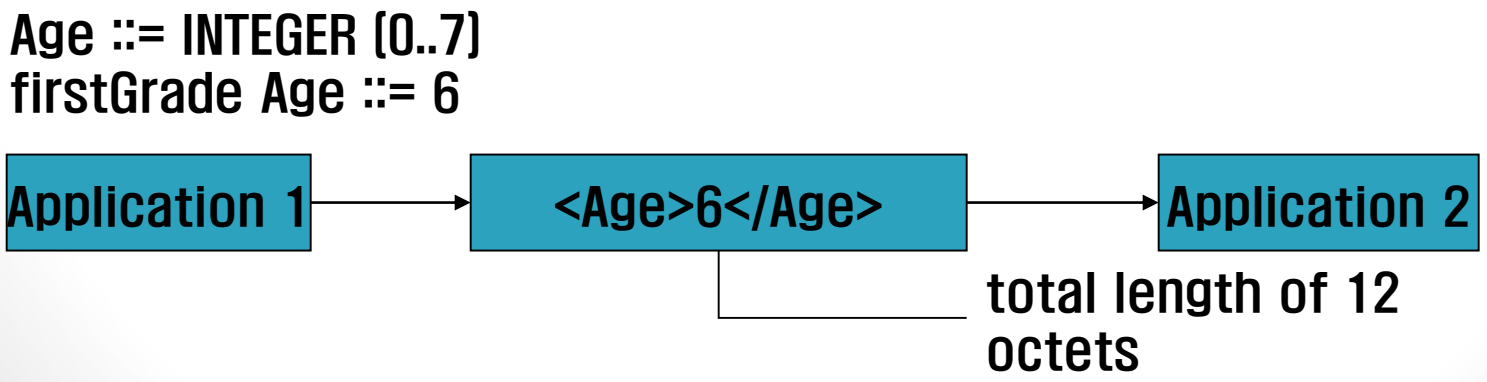
- BER과 마찬가지로 전송되는 데이터가 기기의 종류, 프로그래밍 언어, 어플리케이션 프로그램의 표현방법과는 독립적으로 어떻게 인코드 되어야 하는지를 기술
- BER과는 달리 기본적으로 태그는 전송 안 됨 그리고 length와 value가 종단간에 이미 약속이 되어 있다면 전송되지 않음
- PER은 전송 시 대역폭을 보다 적게 사용하기 위한 목적으로 정의되었으며, 소리와 영상을 전송하는 인터넷, 공중파 통신과 같이 대역폭이 중요한 곳이라면 어느 곳이라도 유용하게 사용

Age ::= INTEGER (0..7)
firstGrade Age ::= 6



XER-XML Encoding Rules

- BER, PER과 마찬가지로 , 전송하기 위한 데이터의 장치의 종류, 프로그래밍 언어 또는 어플리케이션 프로그램의 표현에 대해 독립적
- 축소된 규칙과는 달리 곧바로 읽기 쉽다.
- 좀 더 읽기 쉽게 하는데 목적이 있다.
- 추가 대역폭을 필요로 한다.
- BCN 등 대규모 대역폭을 갖는 환경에 적합하다.



ASN.1/인코딩 규칙 표준 문서

- ITU-T Rec. X.680 | ISO/IEC 8824-1 - Basic ASN.1 Notation
- ITU-T Rec. X.681 | ISO/IEC 8824-2 - Information Object Classes
- ITU-T Rec. X.682 | ISO/IEC 8824-3 - Constraints
- ITU-T Rec. X.683 | ISO/IEC 8824-4 - Parameterization
- ITU-T Rec. X.690 | ISO/IEC 8825-1
  Basic Encoding Rules (BER)
  Canonical Encoding Rules (CER)
  Distinguished Encoding Rules (DER)
- ITU-T Rec. X.691 | ISO/IEC 8825-2
  Packed Encoding Rules (PER)

ASN.1 데이터 타입 (1)

- 프로그래밍 언어와 같이 미리 정의된 몇 가지 타입을 가진다.
- 사용자는 ASN.1이 제공하는 기본 타입 외에 새롭게 정의된 타입을 사용할 수 있다.
- 타입 정의를 위한 일반적인 형식

  <typereference> ::= <type>
- 타입의 이름은 반드시 대문자로 시작해야 한다.
- 대부분의 ASN.1에서 미리 정의된 타입은 대문자로 표기된다.

ASN.1 데이터 타입 (2)

- BIT STRING
  - 비트의 임의의 스트링
- BOOLEAN
  - True/False
- OCTET STRING
  - 8진수 스트링
- IA5STRING
  - ASCII 스트링
- INTEGER
  - 임의의 정수
- REAL
  - 임의의 실수
- NULL
  - 널값

ASN.1 데이터 타입 (3)

- PrintableString
  - 프린트 가능한 스트링
- OCTET IDENTIFIER
  - 객체를 식별하는 정수 값의 연속
- SEQUENCE
  - 서로 다른 ASN.1 데이터 타입들의 순서적 집합
- SEQUENCE OF
  - 동일한 타입의 순서적 집합
- SET
  - 서로 다른 ASN.1 데이터 타입들의 비순서적 집합
- SET OF
  - 동일한 타입의 비순서적 집합

태그(Tags) (1)

- 정의된 모든 타입들은 태그를 지정 받는다.
- 태그는 ASN.1에서 표준으로 지정될 수도 있고, 사용자가 지정할 수도 있고
- 태그는 두개의 부분으로 구성된다
  - Class Universal, Application, Private and Context Specific
  - Number non-negative integer of arbitrary size
- 태그는 PER에 의해 인코딩 되지 않는다

태그(Tags) (2)

- Universal Class
  - 세계적으로 유일하게 지정하는 클래스
- Application Class
  - 어떤 특정 ASN.1 모듈 속에 유일하게 정의하는 클래스
  - 해당 클래스의 데이터 형은 어떤 특정한 응용 속에서만 사용될 태그
- Context-Specific Class
  - 구조 형의 ASN.1 데이터 형 속에 유일하게 지정하는 클래스
  - 해당 클래스 태그는 정의되어 있는 데이터 형 속에 의미를 가지며, 외부에서는 의미를 갖지 않음
- Private Class
  - 상호 합의 되어있는 어떤 특정 기업 및 단체에 속에서 유일하게 지정할 수 있는 클래스

태그(Tags) (3)

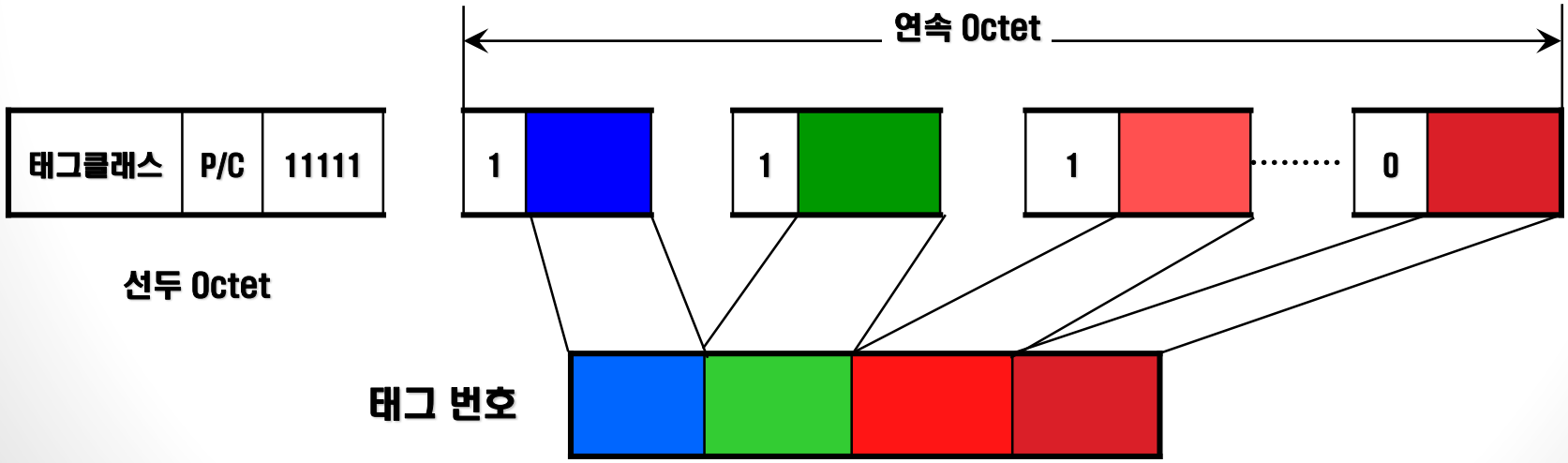
- BER
  - 4개의 구성요소로 구성
    - Identifier Octets
    - Length Octets
    - Contents Octets
    - End of Contents(EOT)
  - Identifier Octets의 Encoding
    - 데이터 값의 형식에 대한 태그 값을 encoding
    - 태그 번호의 범위가 0~30이면 ID Octet은 단일 Octet으로 구성

8비트	7비트	6비트	5비트	4비트	3비트	2비트	1비트

    - 1~5 비트 : 태그 번호
    - 6비트 : P/C (0 : 기본요소, 1: 구조화)
    - 7~8비트 : 태그 클래스(00, 01, 10, 11)

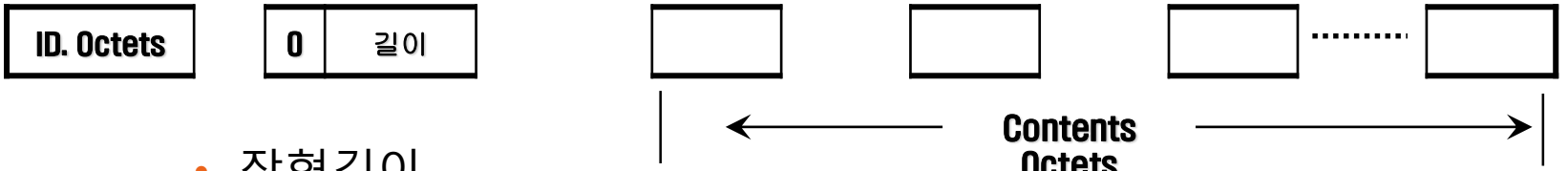
태그(Tags) (4)

- BER-계속
  - Identifier Octets의 Encoding –계속
    - 태그 번호가 31보다 크면 ID. Octet은 하나 이상의 다음 Octet이 따르는 선행 Octet 의로 구성

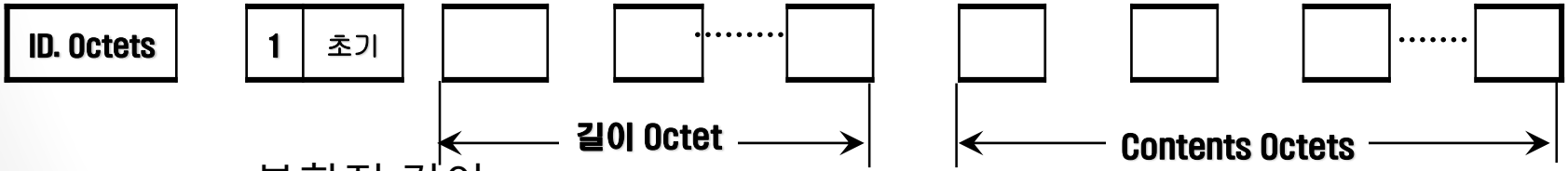


태그(Tags)(5)

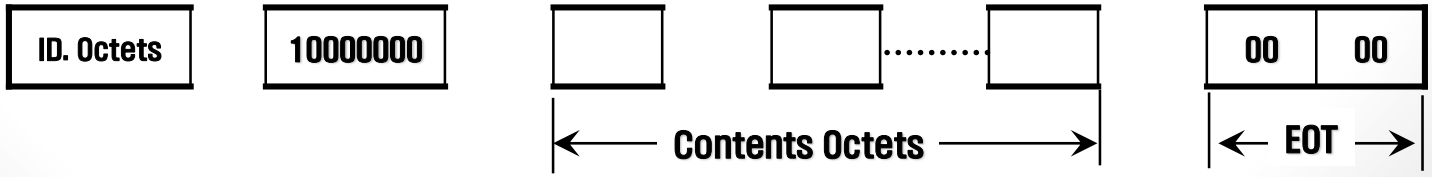
- BER –계속
  - Length Octet
    - 단형길이
      - Contents Octet의 길이를 encode하며, 최대 127개 까지 나타낼 수 있음



    - 장형길이
      - 초기는 길이 Octet의 길이를 나타내며, 길이 Octet은 내용 Octet의 길이를 니타냄



    - 불확정 길이
      - Contents Octets이 EOT에 의해 끝남



태그(Tags)(6)

- BER-계속
  - Contents Octets
    - 0 또는 하나 이상의 Octet으로 구성되며 ASN.1 데이터 값을 encode
      - BOOLEAN 값 encode
        - ID. Octets : 1
        - 부호화 형 : 기본 요소
        - Contents Octet : 단일 Octet
        - 내용 값 : FALSE(0), TRUE(1)
      - 정수 값 encode
        - ID. Octet : 2
        - 부호화 형 : 기본 요소
        - Contents : 하나 이상의 Octet으로 구성되며, 2의 보수로 표시함

태그(Tags)(7)

Type	SNMPv1 protocol	SNMPv2 protocol	ASN.1 Tag	Tag Number	Tag Value
INTEGER/Integer32	✓	✓	UNIVERSAL 2	0x02	0x02
OCTET STRING	✓	✓	UNIVERSAL 4	0x04	0x04
NULL	✓	✓	UNIVERSAL 5	0x05	0x05
OBJECT IDENTIFIER	✓	✓	UNIVERSAL 6	0x06	0x06
SEQUENCE	✓	✓	UNIVERSAL 16	0x10	0x30
IpAddress	✓	✓	APPLICATION 0	0x00	0x40
Counter/Counter32	✓	✓	APPLICATION 1	0x01	0x41
Gauge/Gauge32	✓	✓	APPLICATION 2	0x02	0x42
TimeTicks	✓	✓	APPLICATION 3	0x03	0x43
Opaque	✓	✓	APPLICATION 4	0x04	0x44
Counter64		✓	APPLICATION 6	0x06	0x46

데이터 압축 (1)

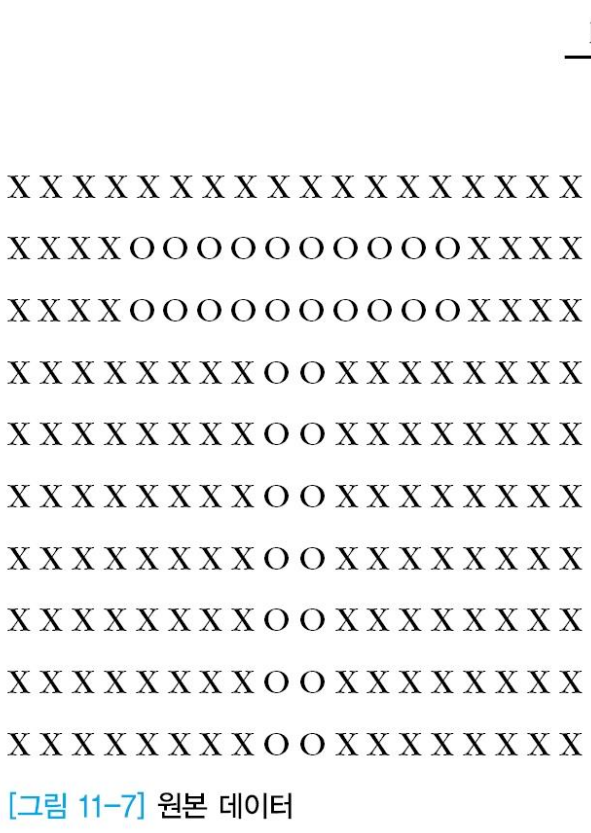
- 데이터 압축과 보안
  - 압축: 전송 데이터의 양을 줄이는 목적으로 사용
  - 암호화: 전송 데이터의 내용을 해석하지 못하도록 하기 위하여 사용
- 대용량 데이터를 압축하여 크기를 줄인 후 전송하는 것이 전송 속도면에서 유리
  - 원래 데이터의 패턴에 영향을 받음
    - 중복이 많으면 압축률이 높음
  - 알고리즘에 따라 압축률이 달라짐
    - 데이터 특성에 맞는 알고리즘 선택이 중요

데이터 압축 (2)

- 연속 문자 압축의 예
  - 원본 데이터: 18 * 10 = 180 바이트
  - 압축 알고리즘
    - 특정 문자가 연속해서 몇 번 나타나는지를 표현

```
struct send_data {
        char pattern; // 임의의 패턴
        short count; // 반복 횟수
}
```

  - 압축 데이터
    - 3 * 19 = 57 바이트
  - 압축 해제(Decompression)
    - 반대의 과정
  - 동일 패턴이 반복되는 환경에서 유리, 그렇지 않으면 오히려 더 커질 가능성



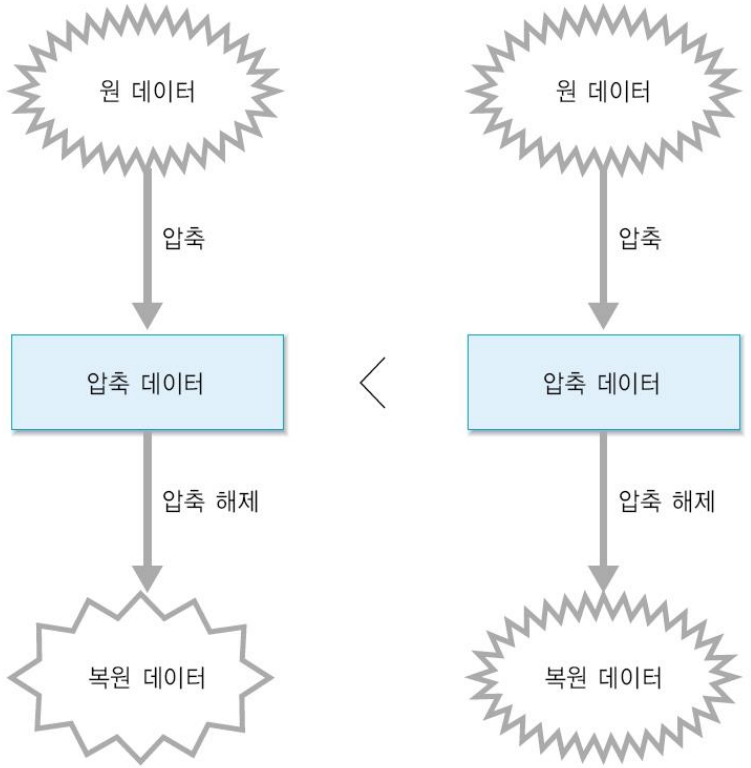
[그림 11-7] 원본 데이터

pattern	count
X	22
O	10
X	8
O	10
X	12
O	2
X	16
O	2
X	16
O	2
X	16
O	2
X	16
O	2
X	16
O	2
X	16
O	2
X	8

[그림 11-8] 압축 데이터

데이터 압축 (3)

- 손실/비손실 압축
  - 비손실(lossless) 압축
    - 압축 과정에서 원래 데이터의 내용을 분실하지 않음
    - 압축 해제 과정을 통해 원래의 데이터를 100% 복원
  - 손실(lossy) 압축
    - 압축 과정에서 원래 데이터의 내용을 부분적으로 분실
    - 압축 해제 과정을 통해 원래의 데이터를 100% 복원할 수 없음
    - 압축 효율을 높이기 위하여 사용



[그림 11-9] 손실 · 비손실 압축