

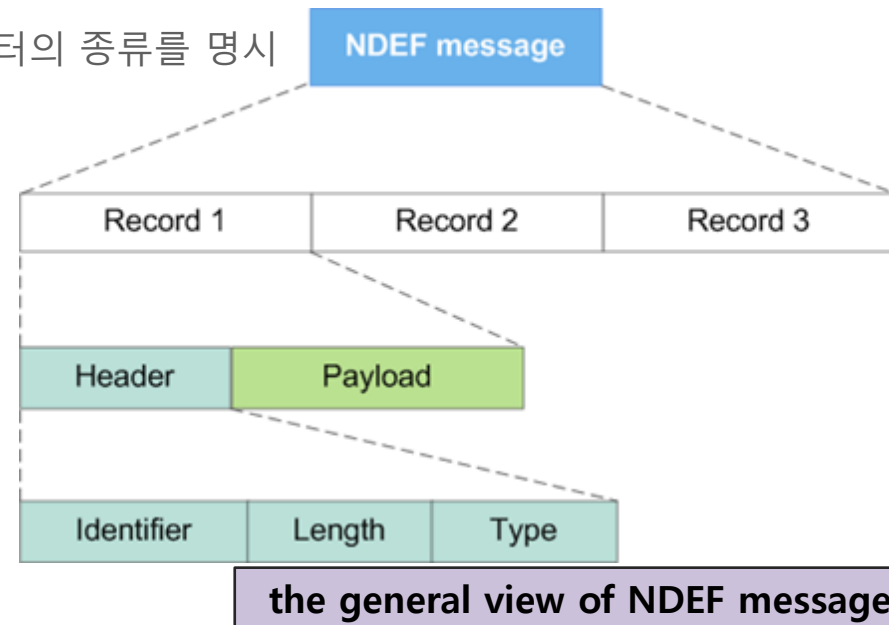
NDEF

NFC Forum Spec

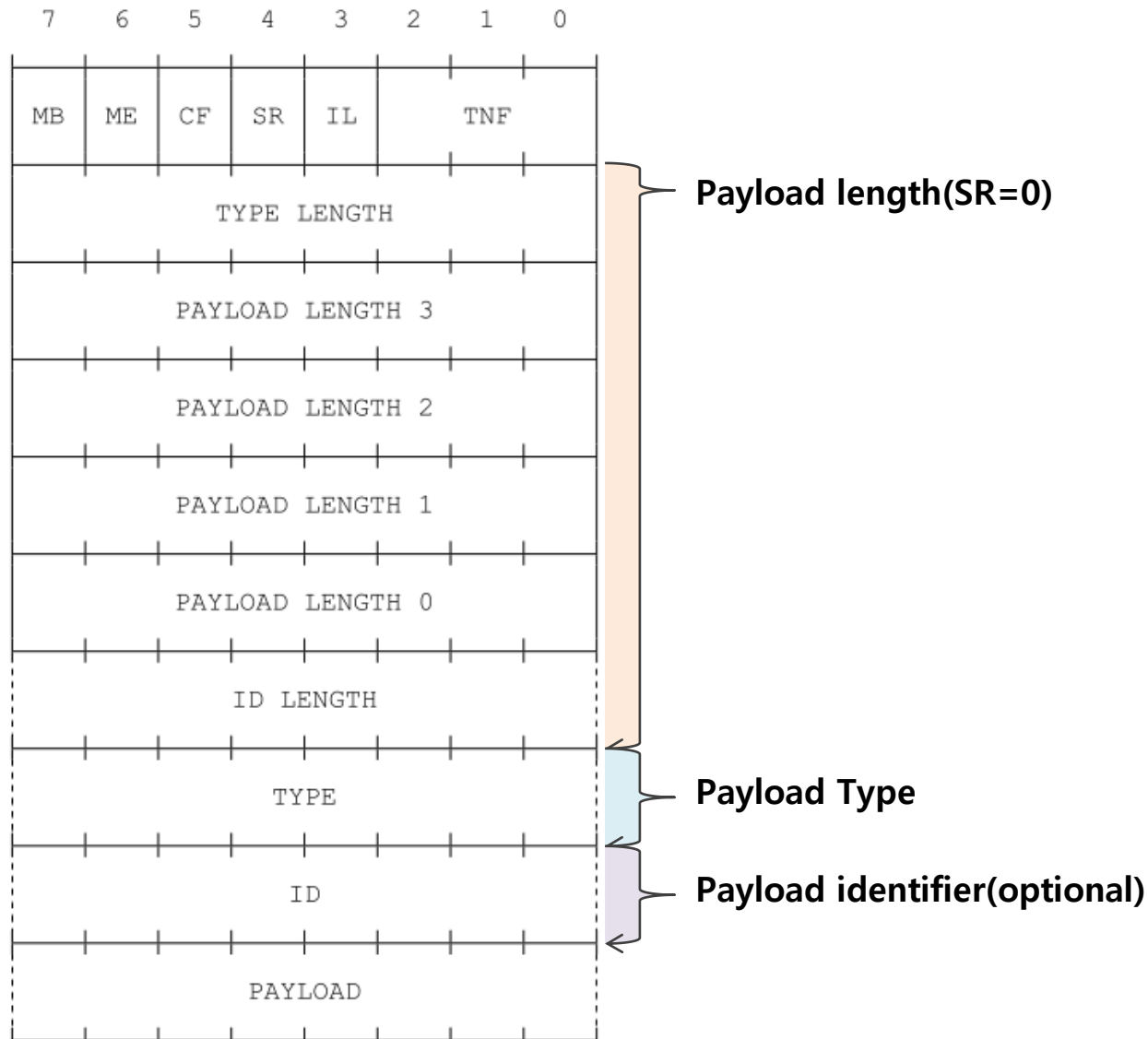


NDEF message and NDEF record

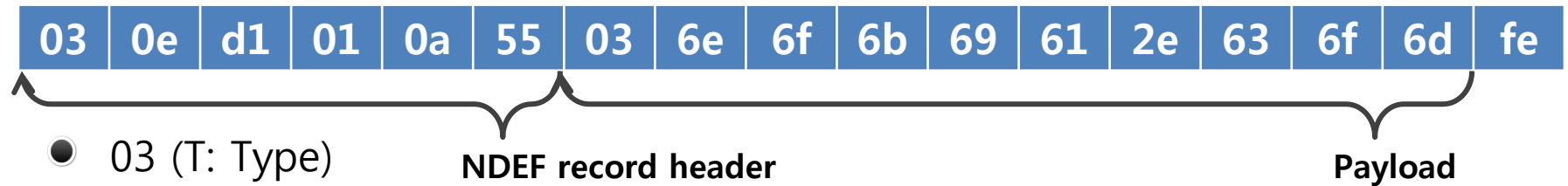
- NDEF는 NFC Forum Data Exchange Format의 약자로서 가벼운 바이너리 메시지 포맷을 지향하며 하나 이상의 애플리케이션에서 정의한 PAYLOAD들을 하나의 메시지 구조에 캡슐화하여 정의
- 하나의 NDEF 메시지는 하나 이상의 NDEF record들을 가지고 있으며 임의의 타입의 PAYLOAD를 전송하는데 목적
- NDEF record는 Payload에 대한 정보를 포함
 - Payload length
 - SR bit 플래그의 값이 1인지 아닌지로 구분
 - 효율적으로 record 경계를 감지
 - Payload type
 - record의 Payload에 포함되어 있는 데이터의 종류를 명시
 - Payload identifier(optional)



NDEF record



NDEF Analysis(Hexa code)-URL(<http://nokia.com>)



- 03 (T: Type)
 - Type of recode
- 0e (L : Lenght)
 - how many bytes are in the payload
- d1 (V : Value)
 - NDEF records are variable length records with a common format illustrated in the figure below.
 - d1 is binary code (11010001)→ IL이 '0'이기 때문에 ID LENGTH가 Layout에 존재하지 않는다.
- 01 → TYPE LENGTH NDEF record header
- 0a → PATLOAD LENGTH ↑
- 55 → TYPE("U")
- 03 → ID(<http://>) - optional
- 6e 6f 6b 69 61 2e 63 6f 6d → Payload (nokia.com)
- fe → Terminator byte

Record Layout (1/2)

1	1	0	1	0	0	0	1
MB	ME	CF	SR	IL	TNF		

- MB = 1
 - Message Begin is true means this is first record in the NDEF message
- ME = 1
 - Message end, means it is last record in the NDEF message, if it is 0 that tells application that more records are ahead
- CF = 0
 - The CF flag is a 1-bit field indicating that this is either the first record chunk or a middle record chunk of a chunked payload

Record Layout (2/2)

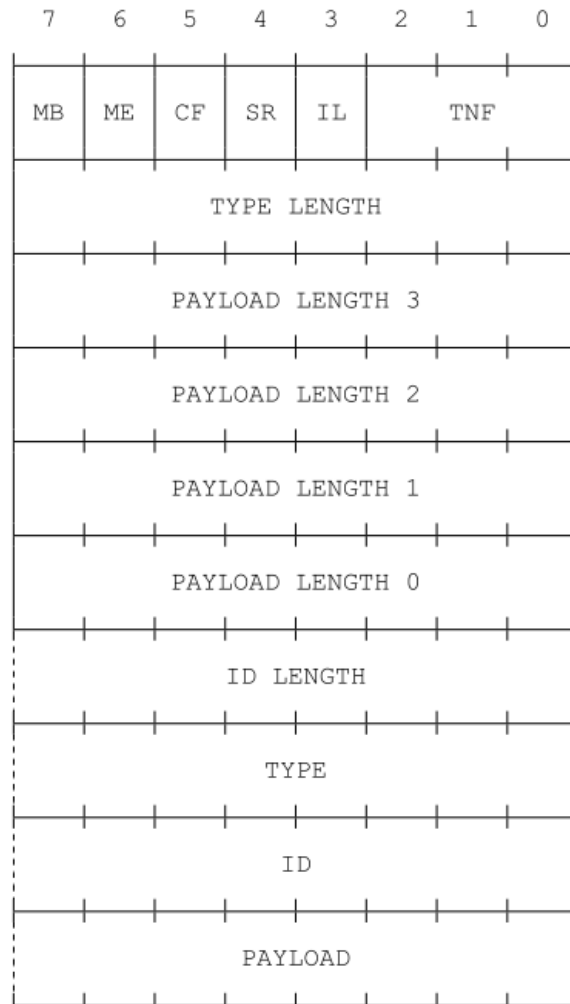
1	1	0	1	0	0	0	1
MB	ME	CF	SR	IL	TNF		

- SR = 1
 - SR stands for short record, if set, that the payload length field is a single octet. This short record layout is intended for compact encapsulation of small payloads which will fit within PAYLOAD fields of size ranging between 0 to 255 octets
- IL = 0 (ID_LENGTH Field와 연관)
 - IL stands for identification length, if set, that the ID_LENGTH field is present in the header as a single octet. **If the IL flag is zero, the ID_LENGTH field is omitted from the record header and the ID field is also omitted from the record**
- TNF = 001
 - The TNF field value indicates the structure of the value of the TYPE field. The value 0x01 (NFC Forum well-known type) indicates that the TYPE field contains a value that follows the RTD type name format defined in the NFC Forum RTD specification

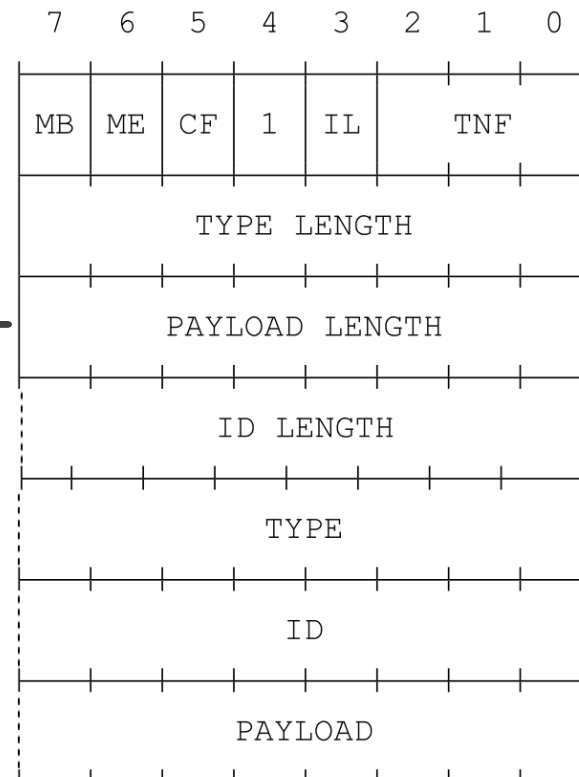
Short-Record Layout(SR = 1)(1/2)

- SR 플래그는 1-bit 필드
- SR = 1
 - PAYLOAD_LENGTH Field는 1 octet
 - PAYLOAD의 길이가 0 ~ 255 octet 사이의 길이가 짧으면 다음과 같이 short record 레이아웃으로 전달
- SR = 0
 - PAYLOAD_LENGTH Field는 4 octet
- short record 레이아웃으로 할지 일반 record 레이아웃으로 할 지는 구현자가 선택하는 것이고 NDEF 파서는 양쪽 다 지원해야 한다.

Short-Record Layout(SR = 1)(2/2)



NDEF Record Layout (SR=0)



NDEF Short-Record Layout (SR=1)

TNF 필드 값 (1/3)

Type Name Format	Value
Empty	0x00
NFC Forum well-known type [NFC RTD]	0x01
RFC 2046에 정의된 Media-type	0x02
RFC 3986에 정의된 절대 URI 경로	0x03
NFC Forum external type [NFC RTD]	0x04
Unknown	0x05
Unchanged	0x06
Reserved	0x07

- TNF Field의 기본값은 없으며 Reserved TNF Field 값은 앞으로 사용하기 위함이며 현재는 사용하지 않아야 한다. NDEF 파서는 알 수 없거나 지원되지 않는 TNF Field 값이 있는 NDEF record를 수신하였을 경우 0x05 (Unknown)으로 변경한다.

TNF 필드 값 (2/3)

- 0x00 (Empty)
 - record와 연관된 타입 혹은 PAYLOAD가 없다는 것을 나타냄
 - TYPE_LENGTH, ID_LENGTH 그리고 PAYLOAD_LENGTH Field는 '0'
 - TYPE, ID 그리고 PAYLOAD Field는 record로 부터 생략
 - TNF 값은 앱에서 더이상의 PAYLOAD가 없을 경우 하나의 NDEF 메시지를 종료한다는 의미로서 빈 record가 필요할 때 사용한다.
- 0x01 (NFC Forum well-known type)
 - TYPE Field에 NFC 포럼 RTD 스펙에서 정의한 RTD 타입 이름 형식을 따르는 값을 포함하고 있음을 나타낸다.
- 0x02 (media-type)
 - TYPE Field에 RFC 2046 에서 정의한 media-type BNF 구조를 따르는 값이 포함되어져 있음을 나타낸다.
- 0x03 (absolute-URI)
 - TYPE Field에 RFC 3986에서 정의한 absolute-URI BNF 구조를 따르는 값이 포함되어져 있음을 나타낸다.

TNF 필드 값 (3/3)

- 0x04 (NFC Forum external type)
 - TYPE Field에 외부 타입 명을 위한 NFC RTD에 정의한 타입 이름 형식을 따르는 값을 포함하고 있음을 나타낸다.
- 0x05 (Unknown)
 - PAYLOAD의 타입을 알 수 없을 때 사용
 - RFC 2046에서 정의한 MIME 중 "application/octet-stream" 과 유사
 - **type_length Field는 '0'이고 TYPE Field는 NDEF record에서 생략**
 - 구현 시 이 값을 사용하는 추가 컨텍스트가 없이 이 타입의 NDEF record를 수신하는 NDEF 파서는 PAYLOAD를 처리하지 않고 저장하는 메커니즘을 제공하는 것을 권고
- 0x06 (Unchanged)
 - 청크된 PAYLOAD들을 포함하고 있는 중간 record 청크와 마지막 record 청크에서 사용
 - 다른 record의 경우에는 사용하지 않아야 한다.
 - **TYPE_LENGTH Field는 '0' 이어야 하고 TYPE Field는 NDEF record에서 생략된다.**
- 0x07
 - 앞으로 어떤 Type을 나타내는 값으로 쓰일지는 모르지만 예약된 값 (Reserved)으로 정의

NDEF Structure(1/2)

● TYPE_LENGTH

- TYPE_LENGTH Field는 8-bit 부호 없는 정수형으로 TYPE Field의 octet 길이를 명시
- TNF가 0x00, 0x05, 0x06일 경우 항상 '0'

● ID_LENGTH

- ID_LENGTH Field는 8-bit 부호 없는 정수형으로 ID Field의 octet 길이를 명시
- 오직 IL이 1로 설정되어있을 경우에만 존재
- ID_LENGTH의 값이 '0'으로 설정되면 ID Field는 NDEF record에서 생략

● PAYLOAD_LENGTH

- PAYLOAD_LENGTH는 부호없는 정수로서 PAYLOAD Field의 octet 길이를 명시
- PAYLOAD_LENGTH Field의 값 길이는 SR 플래그 설정에 의해 결정
 - SR 플래그가 설정되어 있다면 PAYLOAD_LENGTH Field는 8-bit 부호없는 정수를 나타내는 1 octet
 - SR 플래그가 설정되어 있지 않다면 PAYLOAD_LENGTH Field는 32-bit 부호없는 정수형을 나타내는 4 octet이고 MSB-first 순으로 읽는다.
- PAYLOAD_LENGTH Field 값이 0이면 PAYLOAD Field는 NDEF record에서 생략
 - $2^{(32)}-1$ 크기보다 더 큰 PAYLOAD일 경우 청크된 PAYLOAD를 사용한다.

NDEF Structure(2/2)

● TYPE

- TYPE Field의 값은 PAYLOAD의 타입을 설명하는 식별자이다. TYPE Field의 값은 TNF Field 값에 해당하는 구조, 인코딩, 형식을 따라야 한다.
- NDEF 파서는 알 수 없는 TYPE Field 값을 나타내는 TNF Field의 NDEF record가 수신될 경우 마치 TNF Field 값이 0x05(Unknown) 인 것처럼 타입 식별자를 분석해야 한다.
- 타입 식별자는 글로벌적으로 유일하고 안정적으로 유지되고 의미가 오랜 시간 잘 정의되어 사용되는 것을 강력히 권고한다.

● ID

- ID Field의 값은 URI 형식[RFC 3986]의 식별자이다. 유일한 메시지 식별자가 되도록 생성되어야 하며 URI는 절대 혹은 상대적 형식일 수 있다. NDEF는 기본 URI를 정의하지 않으며 사용자 앱에서 상대 경로 형식의 URI를 사용할 경우 실제적이든 가상적이든 기본 URI를 제공해야 한다.
- 중간(middle) 혹은 마지막(terminating) record 청크들은 ID Field를 가지지 않아야 하며 싱글 혹은 첫 번째(initial) record 청크에는 ID Field가 있어야 한다.

● PAYLOAD

- PAYLOAD Field는 NDEF 사용자 앱을 위해 PAYLOAD를 전달한다. PAYLOAD Field를 통해 전달되는 데이터의 내부적 구조에 대해서는 NDEF에서 정의하고 있지 않다.

RTD



Record Types

- Record type의 name은 NDEF applications을 구분하기 위해서 사용됨
- 각각의 Record Type은 NFC Forum 과 third parties 들에 의해 정의됨
- NDEF record header의 TNF Field에 명시

TNF Field Values

Type Name Format	Value
Empty	0x00
NFC Forum well-known type [NFC RTD]	0x01
Media-type as defined in RFC 2046 [RFC 2046]	0x02
Absolute URI as defined in RFC 3986 [RFC 3986]	0x03
NFC Forum external type [NFC RTD]	0x04
Unknown	0x05
Unchanged (see section 2.3.3)	0x06
Reserved	0x07

NFC Forum Well-known Type

- Record의 TNF field 값 → 0x01
- [NFC Forum Well-known Type]은 매우 일반적인 타입들을 정의하고 있음
- *URN으로 정의하는 NFC Forum Well-known Type의 prefix
 - NID(Namespace identifier) → "nfc"
 - NSS(Namespace Specific String) → "wkt:"
- Detail Section
 - NFC Forum Global Type
 - NFC Forum에서 정의하고 관리하는 부분
 - First - character range → <upper>
 - NFC Forum Local Type
 - First - character range → <lower> or <number>

NDEF Record Type	Description	Full URI Reference	Specification Reference
Sp	Smart Poster	urn:nfc:wkt:Sp	NFC Forum Smart Poster RTD
T	Text	urn:nfc:wkt:T	NFC Forum Text RTD
U	URI	urn:nfc:wkt:U	NFC Forum URI RTD
Gc	Generic Control	urn:nfc:wkt:Gc	NFC Forum Generic Control RTD
Hr	Handover Request	urn:nfc:wkt:Hr	NFC Forum Connection Handover Specification
Hs	Handover Select	urn:nfc:wkt:Hs	NFC Forum Connection Handover Specification
Hc	Handover Carrier	urn:nfc:wkt:Hc	NFC Forum Connection Handover Specification
Sig	Signature	urn:nfc:wkt:Sig	NFC Forum Signature RTD

*URN : Uniform Resource Names

NFC Forum Assigned Numbers Register

NFC Forum External Type

- Record의 TNF field 값 → 0x04
- [NFC Forum Well-known Type]은 third parties에서 정의하여 사용
- URN으로 정의하는 NFC Forum External Type의 prefix
 - NID(Namespace identifier) → "nfc"
 - NSS(Namespace Specific String) → "ext:"
- Detail Section
 - third parties에서 사용하는 domain name을 추가하고 third parties에서 관리하는 Type name을 추가
 - EX> "urn:nfc:ext:example.com:f"

RTD Type Names (1/2)

- Requirements for the NFC Forum Well-Known Type Names
- The language used is the ABNF format as defined in *RFC 2234

1. RTD-URI = "urn:nfc:" nfc-nss
2. nfc-nss = wkt-nss / external-nss
3. wkt-nss = wkt-id ":" WKT-type
4. external-nss = external-id ":" external-type
5. wkt-id = "wkt"
6. external-id = "ext"
7. WKT-type = local / global
8. local = (lower / number) *WKT-char
9. global = upper *WKT-char
10. external-type = dns-part ":" name-part
11. dns-part = 1*DNS-char
12. name-part = 1*WKT-char
13. WKT-char = upper / lower / number / other
14. DNS-char = upper / lower / number / "." / "-"

[RFC 2234]

- $A / B == A \text{ OR } B$
- $A*B(\text{element}) == A \sim B$ 회 element를 반복
(default : $A = 0$, $B = \infty$)

RTD Type Names (2/2)

15.upper = "A" / "B" / "C" / "D" / "E" / "F" / "G" / "H" /
"I" / "J" / "K" / "L" / "M" / "N" / "O" / "P" /
"Q" / "R" / "S" / "T" / "U" / "V" / "W" / "X" /
"Y" / "Z"

16.lower = "a" / "b" / "c" / "d" / "e" / "f" / "g" / "h" /
"i" / "j" / "k" / "l" / "m" / "n" / "o" / "p" /
"q" / "r" / "s" / "t" / "u" / "v" / "w" / "x" /
"y" / "z"

17.number = "0" / "1" / "2" / "3" / "4" / "5" / "6" / "7" /
"8" / "9"

18.other = "(" / ")" / "+" / "," / "-" /
":" / "=" / "@" / ";" / "\$" /
"_" / "!" / "*" / "'" / "."

19.Reserved = "%" / "/" / "?" / "#"

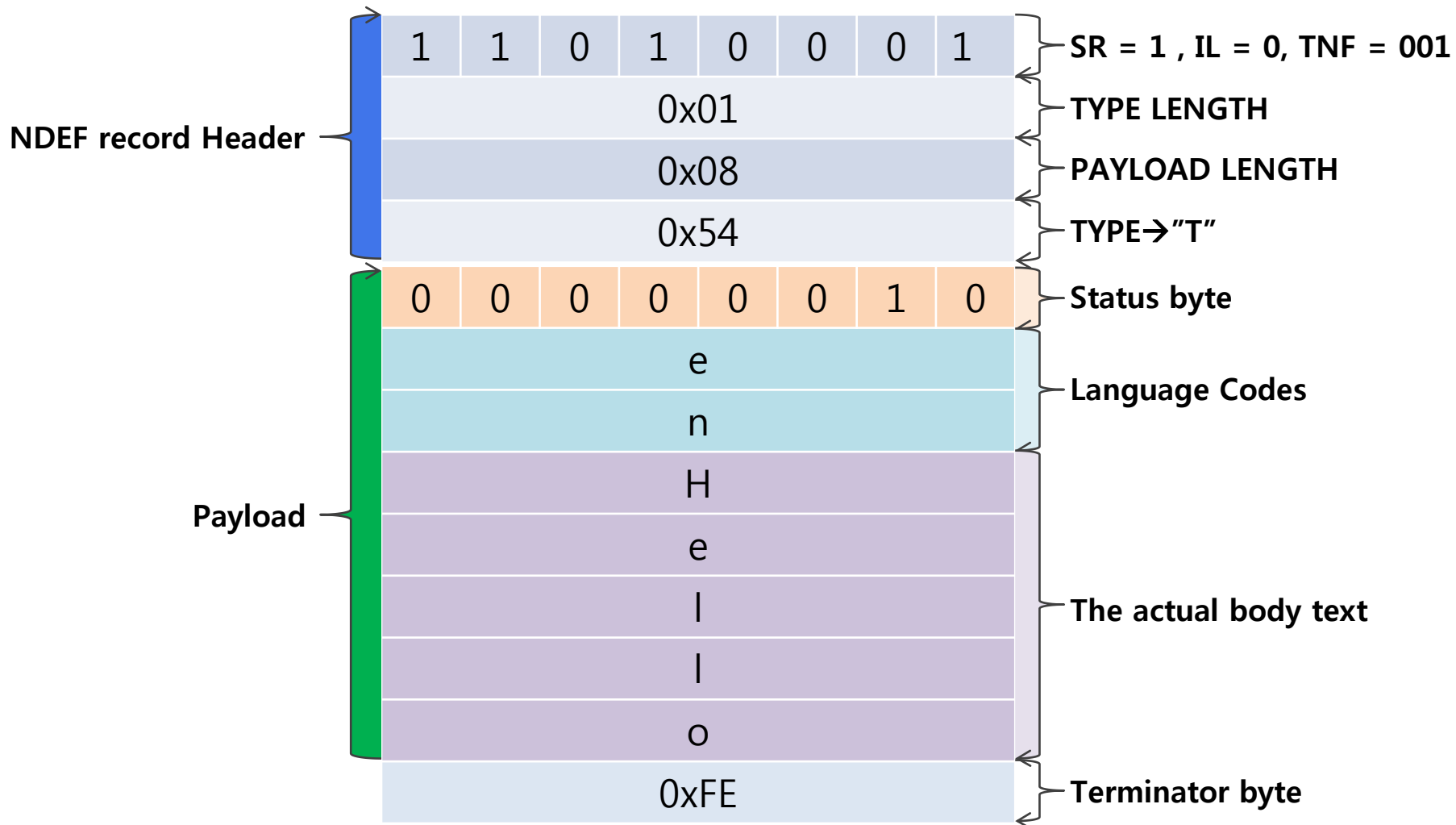
RTD Text

NFC Forum Spec



NDEF Structure

- The NFC Forum Well Known Type [NDEF], [NFC RTD] for the Text record is "T"
- Example ("Hello")



EX) "Hello"

0	0	0	0	0	0	1	0
0: The text is encoded in UTF-8 1: The text is encoded in UTF-16	*RFU	The length of the language code					

↓ 자리수를 결정

e	**ISO/IANA language code
n	

H	The actual body text "Hello"
E	
L	
L	
O	

*Reserved for Future Use(현재는 사용하지 않는 부분으로 0으로 설정)

**<http://www.i18nguy.com/unicode/language-identifiers.html>

RTD URI

NFC Forum Spec



NDEF Structure

- The Well Known Type for an URI record is "U"

- URI identifier code

- URI record의 First Byte는 protocol field를 전달하여 데이터의 크기를 줄인다

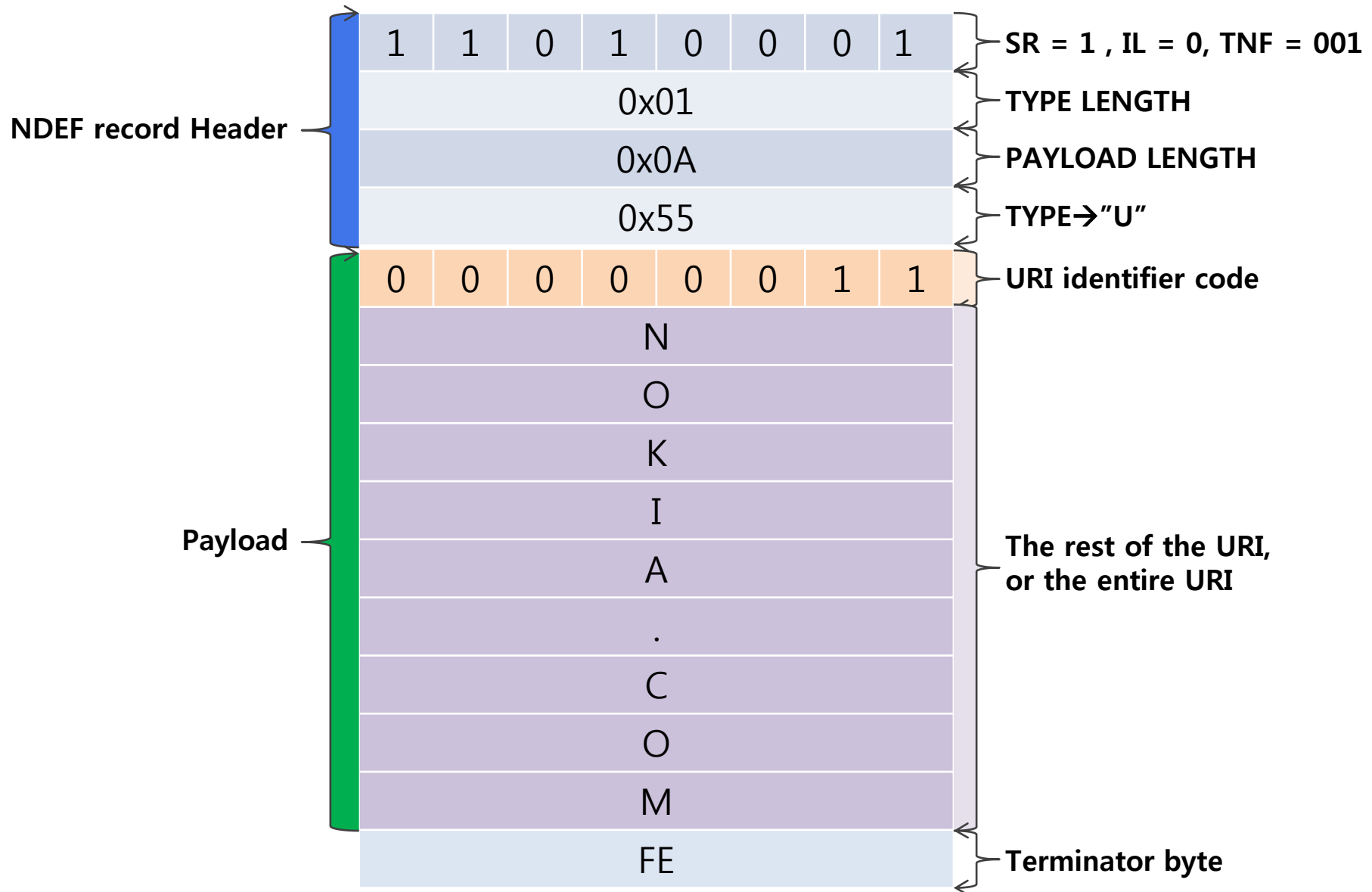
Decimal	Hex	Protocol
0	0x00	N/A. No prepending is done, and the URI field contains the unabridged URI.
1	0x01	http://www.
2	0x02	https://www.
3	0x03	http://
4	0x04	https://
5	0x05	tel:
6	0x06	mailto:
7	0x07	ftp://anonymous:anonymous@
8	0x08	ftp://ftp.
9	0x09	ftps://
10	0x0A	sftp://
11	0x0B	smb://
12	0x0C	nfs://
13	0x0D	ftp://
14	0x0E	dav://
15	0x0F	news:
16	0x10	telnet://
17	0x11	imap:
18	0x12	rtsp://

Decimal	Hex	Protocol
19	0x13	urn:
20	0x14	pop:
21	0x15	sip:
22	0x16	sips:
23	0x17	tftp:
24	0x18	btsp://
25	0x19	bt2cap://
26	0x1A	btgoep://
27	0x1B	tcpobex://
28	0x1C	irdaobex://
29	0x1D	file://
30	0x1E	urn:epc:id:
31	0x1F	urn:epc:tag:
32	0x20	urn:epc:pat:
33	0x21	urn:epc:raw:
34	0x22	urn:epc:
35	0x23	urn:nfc:

※ 0x24~0xFF : RFU

Abbreviation Table

EX) "http://nokia.com"



RTD Smart Poster

NFC Forum Spec



NDEF Structure (1/3)

- The Well-known Type for the Smart Poster record type is "Sp" (in NFC binary encoding: 0x53, 0x70)
- Syntax
 - The URI Record (mandatory)
 - 1개의 URI Record 만 존재
 - 다른 Record들은 URI Record의 metadata
 - The Title record(optional)
 - Text RTD Record를 이용
 - 다른 언어를 사용한 여러 개의 Record가 존재할수 있음
 - URI에 맞는 Title이 표시될 수 있도록 주의를 요함(Phishing)
 - The Icon Record(optional)
 - 다양한 MIME 형태의 image record 포함 가능
 - Device의 가용 범위 내에서 이미지를 표시
 - [PNG],[JPEG] 같은 well-known type 이용을 권장

NDEF Structure (2/3)

- Syntax - continue
 - The Action record(optional)
 - 어떤 서비스를 제공할지 결정
 - Action record가 없는 경우 Device의 추천에 의해서 실행
 - The NFC Local Type Name for the action is "act" (0x61 0x63 0x74).
 - The content is a single byte

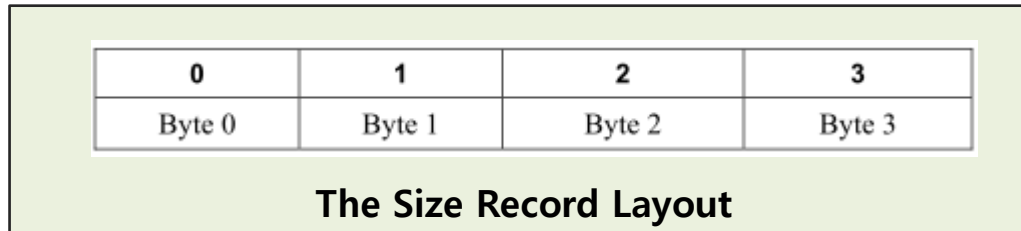
Action Record Values

Value	Action
0	Do the action (send the SMS, launch the browser, make the telephone call)
1	Save for later (store the SMS in INBOX, put the URI in a bookmark, save the telephone number in contacts)
2	Open for editing (open an SMS in the SMS editor, open the URI in an URI editor, open the telephone number for editing).
3..FF	RFU

NDEF Structure (3/3)

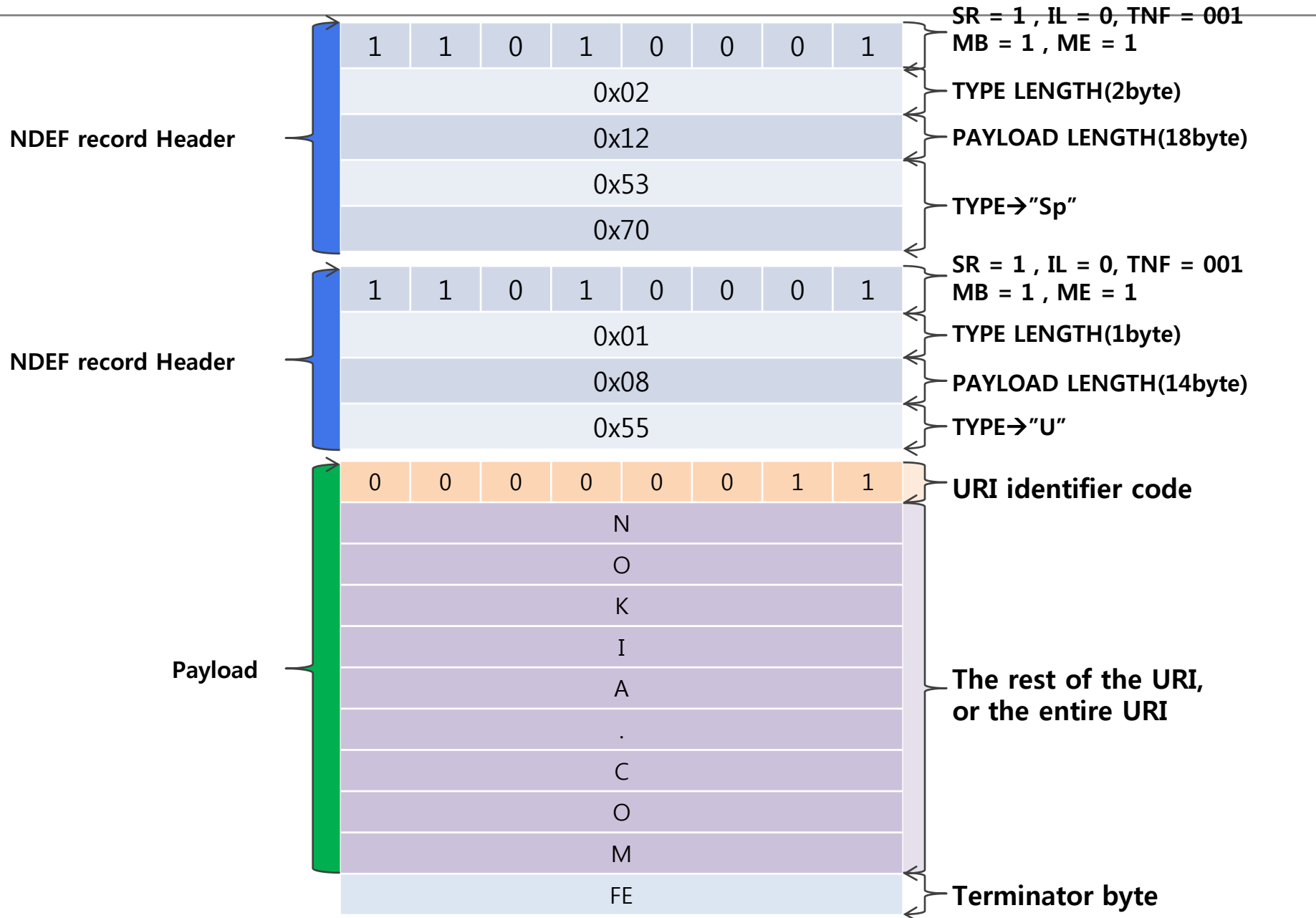
● Syntax - continue

- The Size Record(optional)
 - The Size Record contains a four-byte
 - URI가 외부 독립체로 연결되는 경우 크기를 언급하는 기능 담당
 - Reader device가 URL등을 통해 읽는 content의 크기를 언급해야 하는 경우 활용
 - The Size Record's Local Type Name is "s"



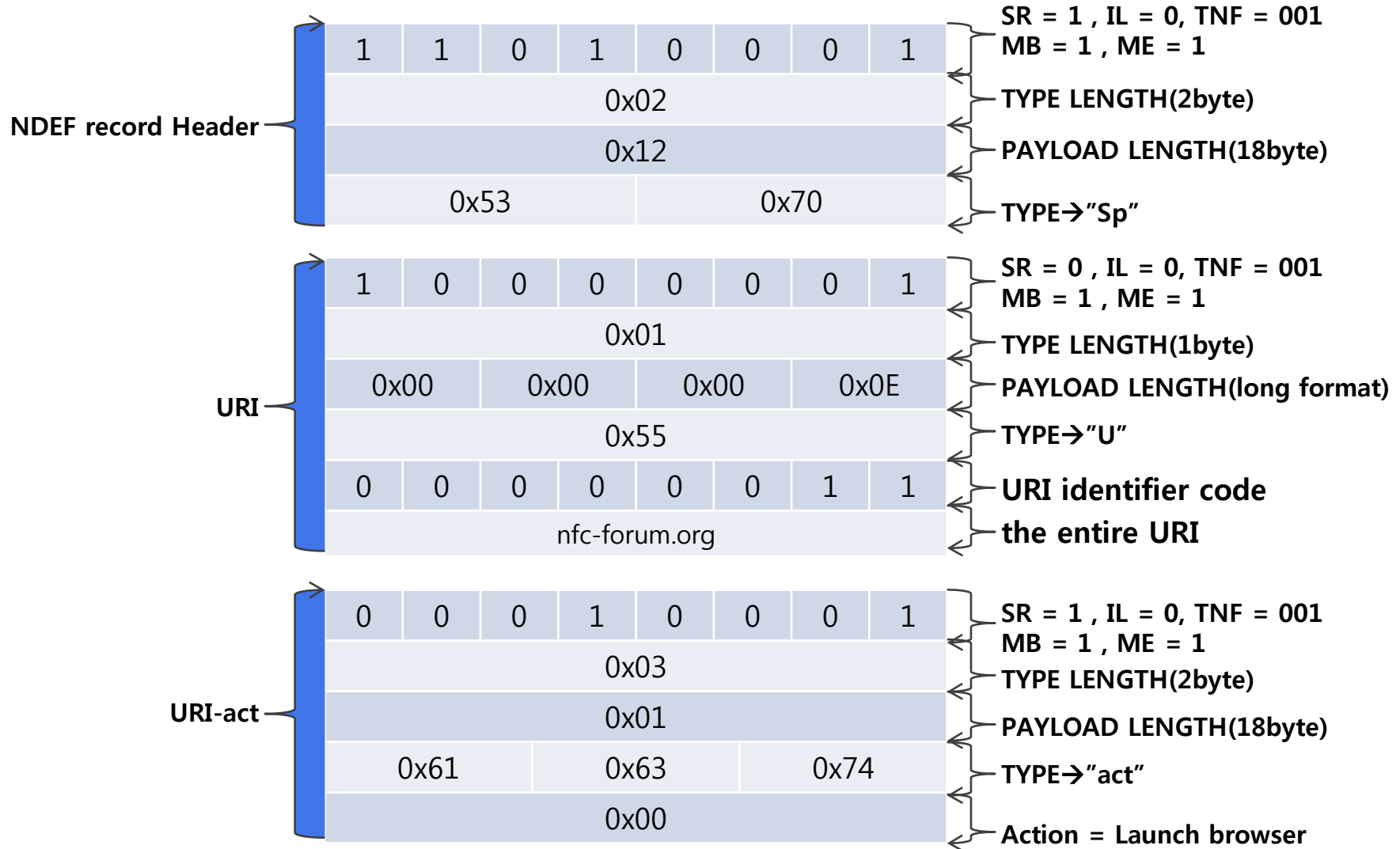
- The Type Record(optional)
 - URI가 외부 독립체로 연결되는 경우 MIME의 타입을 정의
 - Reader device가 URL등을 통해 읽는 MIME의 타입을 언급하여 재생 유/무 판단
 - The Local Type Name for the Type Record is "t"

EX) Simple URI on a Tag



EX) Complex URI on a Tag (1/2)

- A URI with a Title and Launching a Browser



EX) Complex URI on a Tag (2/2)

