

ASN.1을 이용한 가변 길이 메시지 표현 방법[†]

(An approach to define variable length messages using ASN.1)

백하은* 강성원** 김진규*** 김정민****
(Haeun Baek) (Sungwon Kang) (Jingyu Kim) (Jungmin Kim)

권구형[†] 김상수[‡]
(Koohyung Kwon) (Sangsoo Kim)

요 약 Variable Message Format(VMF)는 제한된 대역폭을 가지는 통신 환경에서 전송 정보를 효율적으로 교환하기 위하여 개발된 통신 메시지 표준으로, 지시자의 사용을 통해 단지 요구된 정보만을 전송할 수 있도록 메시지 길이 및 구조를 가변적으로 정의할 수 있도록 설계되었다. 그러나 이러한 가변성은 메시지를 효과적으로 분석하고 그 의미를 추출하는 것을 어렵게 할 뿐만 아니라, 메시지의 추가나 변경 시에 메시지 처리와 관련된 소프트웨어를 매번 수정하고, 이를 사용하는 모든 기기에 다시 배포해야 하는 번거로움이 있다. 본 논문에서는 국제 표준의 정형 기법인 ASN.1을 이용하여 VMF 메시지를 표현하는 체계적인 방법을 제안한다. 비트열 형태의 VMF 메시지를 구조적인 ASN.1 자료 구조로 표현함으로써 프로토콜 설계자는 고 수준에서 메시지의 구조와 그 값을 다룰 수 있으며, ASN.1을 지원하는 다양한 도구들을 사용할 수 있다. 제안하는 방법은 VMF 메시지 세그먼트들에 대해 템플릿을 정의하고, 이를 조합하여 다양한 VMF 메시지를 표현한다.

키워드 가변 길이 메시지, VMF, 추상 구문 표기법, ASN.1

Abstract Variable Message Format(VMF) is a communications protocol that allows computers to exchange tactical military information over low-data-rate bearers. VMF is designed to define message length and structure variably in order to allow the user to transmit only the relevant information by using indicators. However, flexibility of messages makes it difficult to analyze messages structure and figure out meanings. Furthermore, whenever messages are added or modified, message processing software should be updated and distributed to systems. In this paper, we propose a systematic approach to defining VMF messages that uses the international standard notation ASN.1. By describing VMF messages in ASN.1, protocol architects can deal with VMF messages accurately and conveniently, and use various ASN.1 tool. This paper develops ASN.1 description templates for VMF message segments, and presents an approach to describing VMF messages in ASN.1 by combining the templates.

Key words Variable length message, Variable Message Format, Abstract Syntax Notation One

[†] 본 연구는 국방과학연구소에서 지원하는 ‘실시간 전송정보 처리를 위한 최적화 방안 연구’ 위탁연구과제로 수행되었습니다.

* 학생회원 : 한국과학기술원 전산학과, haeun@kaist.ac.kr

** 중신회원 : 한국과학기술원 전산학과 교수,
sungwon.kang@kaist.ac.kr

*** 학생회원 : 한국과학기술원 전산학과, jinlooks@kaist.ac.kr

**** 학생회원 : 한국과학기술원 전산학과, jmin11@kaist.ac.kr

[†] 비 회원 : 국방과학연구소, koohyung@add.re.kr

[‡] 비 회원 : 국방과학연구소, plus@add.re.kr

1. 서론

Variable Message Format(VMF)는 제한된 대역폭을 가지는 전송 통신 환경에서 실시간/근실시간으로 전송 정보를 교환하기 위해 제안된 비트 단위의 가변길이 메시지에 대한 디지털 정보 교환 표준이다[1]. VMF 메시지는 비트 열로 표현되며, 메시지 내의 인접한 데이터 필드들은 공백 문자 없이 연결된다. 또한 지시자의 값에 따라 뒤따르는 필드의 존재 및 반복 여부가 결정되도록 함으로써, 정보 요청 시 단지 요구된 의미 있는 정보만을 전송할 수 있도록 하는 유연성을 제공한다.

그러나 이러한 특징은 지시자들을 순서대로 해석해 나가지 않고서는 VMF 메시지의 길이와 구조의 예상을 불가능하게 할 뿐만 아니라, 메시지를 효과적으로 분석하고 그 의미를 추출하는 것을 어렵게 한다. 또한 신규 메시지의 추가나 기존 메시지의 변경이 이루어 질 때에는, 이들을 정확하고 신속하게 지상 전송 통신 환경에 반영하기 위해 파서 소프트웨어를 비롯한 전송 정보 처리 소프트웨어를 매번 수정하고, 이를 사용하는 모든 기기에 다시 배포해야 하는 어려움이 있다.

ASN.1은 추상 구문 표기법(Abstract Syntax Notation One)의 약어로, 국제 전기 통신 연합 전기 통신 표준화 부문(ITU-T)에서 정의한 네트워크 상에서 전송되는 데이터를 기술하기 위한 정형기법이다[4]. ASN.1은 데이터를 정수 타입(Integer), 비트열 타입(Bit stream) 등의 데이터 타입 및 데이터 값에 따라 정의하고, 다양한 데이터 타입을 조합시켜 복잡한 데이터 구조를 기술할 수 있도록 한다. ASN.1은 여러 분야에서

활용되고 있으며, 자동화 도구들이 많이 개발되어 있다[4-8].

VMF 메시지를 ASN.1으로 표현하게 되면 비트열 형태의 VMF 메시지를 구조적인 데이터 구조로 기술할 수 있으며, 이에 따라 고 수준에서 VMF 메시지의 구조와 그 값을 정확하고 편리하게 다룰 수 있다. 또한, 메시지의 추가나 변경 시에 소프트웨어를 새로 만들거나 수정할 필요 없이 해당 메시지 명세만 ASN.1으로 표현하여 주기만 하면 TTCN-3 기반의 테스트 도구와 같이 ASN.1을 지원하는 다양한 도구들을 사용할 수 있다.

본 논문에서는 VMF 메시지 명세를 국제 표준의 정형기법인 ASN.1을 이용하여 표현하는 체계적인 방법을 제안한다. 이를 위해, 먼저 VMF 메시지의 가변성을 이루는 메시지 명세 세그먼트들에 대해 ASN.1으로 표현하는 템플릿을 정의한다. 이어서 앞서 정의한 템플릿을 조합하여 하나의 VMF 메시지 명세를 ASN.1으로 표현하는 방법을 제안한다. 메시지를 세그먼트 단위로 분할하여 각각에 대해 템플릿을 적용함으로써, VMF 메시지의 구조나 길이에 영향을 받지 않고 다양한 VMF 메시지 각각에 대해서 ASN.1 표현을 정의할 필요 없이 템플릿의 조합으로 쉽게 표현할 수 있다.

본 논문의 구성은 다음과 같다: 2장에서는 VMF 메시지의 구조와 ASN.1 표기법에 대해 논의한다. 3장에서는 VMF 메시지를 ASN.1으로 표현하는 방법을 제안한다. 4장에서는 3장에서 제안하는 방법을 이용하여 특정 메시지 명세 예제를 ASN.1으로 표현하고, 마지막으로 5장에서는 결론을 도출한다.

2. 배경 연구

이 장에서는 VMF 메시지를 ASN.1으로 표현 하는데 있어 주목할 만한 VMF 메시지 구조와 VMF 메시지 명세 표현에 이용되는 ASN.1 문법에 대해 정리한다.

2.1 VMF 메시지의 구조

VMF 메시지는 필드들의 나열로, 정보필드와 지시자라 불리는 문법필드의 조합으로 구성된다. 여기서 정보 필드는 일반 자료 정보를 담고 있는 필드로, 해당 필드 의미와 기능에 따라 각기 다른 길이의 비트 열로 구성된다. 문법 필드는 정보 필드의 존재 혹은 반복 여부를 나타내는 필드로 1 비트의 길이를 가진다. 문법 필드는 자료 요소의 존재 여부를 나타내는 존재 지시자(Presence Indicator)와 자료 요소의 반복 여부를 나타내는 반복 지시자(Recurrence Indicator)로 나뉘며, 지시자의 종류에는 FPI(Field Presence Indicator), GPI (Group Presence Indicator), FRI (Field Recurrence Indicator), 그리고 GRI (Group Recurrence Indicator) 가 있다[1, 2, 3].

VMF 메시지의 가변성은 지시자와 지시자의 값에 따른 정보 필드의 존재 및 반복 여부에 의해 구현된다. 지시자의 값에 따라 뒤따르는 필드가 포함되지 않을 수도 있고, 여러 번 반복되어 포함될 수도 있다. 그림 1은 각 지시자로부터 정의되는 VMF 메시지의 가변성 명세 메커니즘을 설명한다. 그림 1에서 ‘[]’기호는 선택적(optional) 필드를 나타낸다. ‘val(I)’는 지시자 I의 값을 의미하며, ‘0’ 또는 ‘1’의 값을 가질 수 있다. ‘(F)+’는 필드 F가 하나 또는 그 이상 반복됨을 나타내며, ‘(F)*’는 필드 F가 0회 또는 그 이상 반복됨을 나타낸다[2].

(A) Field Presence Indicator (FPI)

FPI 다음에 연속되는 하나의 자료요소의 존재 여부를 정의한다.

사용: FPI + [DATA_FIELD]

→ If val(FPI) = 1 then DATA_FIELD must exist.

→ If val(FPI) = 0 then DATA_FIELD must not exist.

(B) Group Presence Indicator (GPI)

GPI 다음에 여러 개의 필드로 구성된 하나의 그룹의 존재

여부를 정의한다. 사용: GPI + GROUP*

→ If val(GPI) = 1 then GROUP* must exist.

→ If val(GPI) = 0 then GROUP* must not exist.

(C) Field Recurrence Indicator (FRI)

FRI 다음에 하나의 자료요소가 반복됨을 정의한다.

사용: (FRI + [DATA_FIELD])+

→ If val(FRI) = 1 then DATA_FIELD must exist and ‘FRI + [DATA_FIELD]’ must be repeated.

→ If val(FRI) = 0, then its associated DATA_FIELD must not exist and ‘FRI + [DATA_FIELD]’ must not be repeated any longer.

(D) Group Recurrence Indicator (GRI)

GRI 다음에 하나의 반복 그룹 구조가 반복됨을 정의한다.

사용: (GRI + GROUP*)+

→ If val(GRI) = 1 then GR* must exist and ‘(GRI + GROUP*)’ must be repeated.

→ If val(GRI) = 0, then GR* must not exist and ‘(GRI + GROUP*)’ must not be repeated any longer.

(where GROUP = DATA_FIELD | FPI + [DATA_FIELD] | GPI + GROUP* | (FRI + [DATA_FIELD])+ | (GRI + GROUP*)+)

그림 1. VMF 메시지의 가변성 명세 메커니즘

본 논문에서는 VMF 메시지의 가변성을 구현 하는 지시자를 기준으로 VMF 메시지 명세 세그먼트를 정의하고, 주어진 VMF 메시지 명세를 구성하는 각 세그먼트들에 대해 ASN.1 표현 템플릿을 정의 및 적용함으로써 ASN.1 표현을 완성한다.

2.2 ASN.1 표기법

ASN.1 표기법은 정수나 비트열과 같은 기본적인 데이터 타입에서부터 집합(Set)이나 순서열(Sequence)과 같은 복합적인 데이터 타입에 이르기 까지 다양한 기본 데이터 타입을 정의함은 물론, 이들 기본 데이터 타입을 조합하여 새로운 데이터

타입을 정의할 수 있도록 한다. 따라서 프로토콜 설계자는 ASN.1을 이용함으로써 통신 매체를 통해 교환되는 비트나 바이트와 같은 형태의 정보를 구조적인 데이터 형태로 기술할 수 있으며, 고 수준에서 정보의 구조와 값을 다룰 수 있다.

컴퓨터 간에 교환되는 정보는 정보의 종류 및 형태를 나타내는 타입과 정보의 내용이자 실체인 값으로 구성된다. ASN.1은 정보 데이터를 타입과 값으로 정의하며, 각각에 대한 표기법을 제공한다. 새로운 데이터 타입을 정의하는 것을 타입 배정, 데이터 타입의 객체에 값을 지정하는 것을 값 배정이라고 한다[8].

예를 들어 '나이 25세'란 정보를 ASN.1으로 표현한다면, 다음과 같은 타입 배정과 값 배정을 생각할 수 있다.

```
타입 배정문) Age ::= Integer
값 배정문) myAge Age ::= 25
```

위의 타입 배정문은 나이를 표시하기 위한 새로운 데이터 타입 Age에 먼저 정의되어 있는 데이터 타입인 Integer를 배정하는 것을 나타낸다. 이로써 나이를 표현하기 위한 데이터 타입으로 Age를 참조하는 것이 가능해진다. 이어지는 값 배정문은 myAge라는 이름의 변수에 Age라는 데이터 타입의 값 25를 배정하는 것을 나타내며, 이로써 나이 값 25를 myAge로 참조하는 것이 가능하게 된다.

본 논문에서는 ASN.1의 SEQUENCE 타입과 CHOICE 타입을 조합하여 새로운 데이터 타입을 정의함으로써 VMF 메시지를 표현한다.

SEQUENCE 타입: 0개 이상의 ASN.1 데이터 타입들의 나열을 나타내는데 사용되며, 컴포넌트들의 나열 순서가 의미를 가진다[4].

CHOICE 타입: 하나 이상의 데이터 타입들의 공유형으로, 여러 개의 컴포넌트들 중 하나를 선택

할 수 있도록 한 데이터 타입이다. 후보 컴포넌트들 중 하나를 선택해내기 위해서는 컴포넌트들의 데이터 타입들이 상호 식별 가능해야 한다. 이들 중 같은 데이터 타입을 가지는 후보 컴포넌트가 있는 경우에는 명시적으로 태그를 붙여 구분할 수 있도록 한다[4].

ASN.1은 특정 장치, 데이터 표현방식, 프로그래밍 언어, 하드웨어 플랫폼에 종속되지 않고 데이터를 정의할 수 있다. ASN.1을 지원하는 도구들은 거의 모든 운영체제에서 동작가능하며, JAVA, C, C#에서부터 COBOL에 이르기까지 다양한 프로그래밍 언어로의 변환을 지원한다. ASN.1은 여러 분야에서 활용되고 있으며, 이를 위한 자동화 도구들이 많이 개발되어 있다[4-8].

3. ASN.1을 이용한 VMF 메시지의 가변성 표현 방법

이 장에서는 ASN.1을 이용하여 VMF 메시지를 표현하는 체계적인 방법을 제안한다. VMF 메시지는 메시지 내의 자료 항목들의 구성을 정의한 메시지 양식 표준인 메시지 명세와 실제 전송 정보 데이터를 담고 있는 비트열 형태의 VMF 메시지 인스턴스(instance)로 구분할 수 있다. 이에 따라, VMF 메시지 명세는 ASN.1의 타입 배정으로, VMF 메시지 인스턴스는 ASN.1의 값 배정으로 각각 대응시켜 표현할 수 있다. VMF 메시지 인스턴스를 ASN.1으로 표현하는 것은 해당 메시지 인스턴스에 대한 메시지 명세의 ASN.1 타입 배정이 주어진다면, 이에 대해 값을 넣어주어 값 배정문으로 작성하는 것으로 자명하다.

따라서, 이 장에서는 VMF 메시지 명세를 ASN.1의 타입 배정문으로 표현하는 방법을 제안한다. 이를 위해, 먼저 VMF 메시지 명세 세그먼트

들에 대해 ASN.1으로 표현하는 템플릿을 정의한다. 그리고 특정 VMF 메시지 명세가 주어지면, 앞서 정의한 템플릿을 이용하여 ASN.1으로 표현하는 방법을 제시한다.

3.1 VMF 메시지 명세 세그먼트에 대한 ASN.1 표현 템플릿 정의

제 2장에서 언급한 바와 같이, VMF 메시지의 가변성은 네 개의 지시자에 의해 구현된다: FPI,

표 1. VMF 메시지 명세 세그먼트에 대한 ASN.1 표현 템플릿

VMF 메시지 명세 세그먼트					ASN.1 표현 템플릿
(A) 세그먼트-FPI					<pre> SEG-FPI ::= CHOICE { noPresence [0]FPI-N, presence [1]FPI-P } FPI-N ::= SEQUENCE { fpi BIT STRING('0'B) } FPI-P ::= SEQUENCE { fpi BIT STRING('1'B), idx1-1 BIT STRING(SIZE(n)) } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
1.	FPI	1			
1.1	data	n			
(B) 세그먼트-GPI					<pre> SEG-GPI ::= CHOICE { noPresence [0]GPI-N, presence [1]GPI-P } GPI-N ::= SEQUENCE { gpi BIT STRING('0'B) } GPI-P ::= SEQUENCE { gpi BIT STRING('1'B), Gn SEG-Gn } SEG-Gn ::= SEQUENCE { idx1-1 BIT STRING(SIZE(n1)), idx1-2 BIT STRING(SIZE(n2)) } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
1.	GPI	1			
1.1	data1	n1	Gn		
1.2	data2	n2	Gn		
(C) 세그먼트-FRI					<pre> SEG-FRI ::= CHOICE { noRecurrence [0]FRI-N, recurrence [1]FRI-P } FRI-N ::= SEQUENCE { fri BIT STRING('0'B), idx1-1 BIT STRING(SIZE(n)) } FRI-P ::= SEQUENCE { fpi BIT STRING('1'B), idx1-1 BIT STRING(SIZE(n)), seg-fri SEG-FRI } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
1.	FRI	1			
1.1	data	n			
(D) 세그먼트-GRI					<pre> SEG-GRI ::= CHOICE { noRecurrence [0]GRI-N, recurrence [1]GRI-P } GRI-N ::= SEQUENCE { gri BIT STRING('0'B), Rn SEG-Rn } GRI-P ::= SEQUENCE { gri BIT STRING('1'B), Rn SEG-Rn, seg-gri SEG-GRI } SEG-Rn ::= SEQUENCE { idx1-1 BIT STRING(SIZE(n1)), idx1-2 BIT STRING(SIZE(n2)) } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
1.	GRI	1			
1.1	data1	n1		Rn	
1.2	data2	n2		Rn	

GPI, FRI, 그리고 GRI. 이에 따라, VMF 메시지를 각 지시자를 기준으로 지시자와 지시자에 의해 그 존재 및 반복 여부를 제한 받는 정보 필드들을 하나의 세그먼트 단위로 구분 지어 분할해낼 수 있다. 표 1은 각 VMF 메시지 명세 세그먼트에 대한 ASN.1 표현 템플릿을 정의한 것이다.

VMF 메시지 명세로부터 VMF 메시지 인스턴스를 생성하게 되면 지시자의 값에 따라 뒤따르는 필드들의 구성이 달라진다. 이러한 가변성은 ASN.1의 CHOICE 타입을 이용하여 표현할 수 있다. 표 1에서 정의한 ASN.1 표현 템플릿에서는 태그 값에 의해 각 세그먼트들의 첫 번째 필드인 지시자의 데이터 값이 '0'인 경우에는 'noPresence' 또는 'noRecurrence' 컴포넌트가, '1'인 경우에는 'presence' 또는 'recurrence' 컴포넌트가 선택된다. 각 컴포넌트들은 필드들의 나열로 SEQUENCE 타입으로 나타낸다.

지시자 GPI와 GRI에 의해 그 존재 및 반복 여부를 제한 받는 하위 필드들의 그룹은 표 1의 (B)세그먼트-GPI의 SEG-Gn과 같이 그룹 코드 또는 반복 코드 이름으로 새로운 데이터 타입을 선언하고, 여기에 SEQUENCE 타입을 배정하여 정의한다.

반복 지시자 FRI와 GRI에 대한 세그먼트의 경우, 지시자의 값이 '1'인 경우에 전체 구조가 재귀적으로 반복된다. 따라서 표 1의 (C)세그먼트-FRI의 seg-fri와 같이 'presence' 또는 'recurrence' 컴포넌트의 마지막에 세그먼트의 데이터 타입을 가지는 컴포넌트를 두어 그 구조가 재귀적으로 반복되어 나타날 수 있도록 정의한다.

정보 필드들은 ASN.1 이 제공하는 기본 데이터 타입인 BIT STRING 타입으로 나타내며, 'SIZE' 키워드를 이용하여 각 필드의 비트 수를 정의한다. 표 1에서 굵게 표시된 구문은 주어진 메시지 명세의 구성에 따라 다르게 정의되는 내용들이다.

3.2 특정 VMF 메시지 명세를 ASN.1으로 표현하는 방법

ASN.1으로 표현하고자 하는 VMF 메시지 명세가 주어지면, 3.1절에서 정의한 표 1의 템플릿들을 조합하여 하나의 ASN.1 모듈로 표현할 수 있다. 하나의 VMF 메시지 명세를 ASN.1으로 표현하는 방법은 다음의 세 단계로 이루어진다.

- 단계 1) VMF 메시지 명세의 계층 구조 분석
- 단계 2) VMF 메시지 명세를 메시지 명세 세그먼트 단위로 분할
- 단계 3) ASN.1 표현 템플릿의 적용

단계 1) VMF 메시지 명세의 계층 구조 분석

VMF 메시지는 필드들의 나열로 필드들 간에 계층 구조를 가지며, 따라서 이를 트리 구조로 나타낼 수 있다. 지시자는 연속되는 하위 필드의 존재 혹은 반복 여부를 나타내는 필드로, 지시자에 의해 존재 혹은 반복 여부를 제한 받는 하위 필드들을 자식(Child) 노드로 가지게 된다. 각 하위 필드 역시 트리 구조로 나타낼 수 있다.

GPI는 여러 개의 필드로 구성된 하나의 그룹이 존재하는지 여부를 나타내는 지시자로, GPI에 의해 그 존재 여부를 제한 받는 하위 필드들은 GPI와 같은 그룹 코드를 가진다. FRI와 GRI는 반복 지시자로 하나 이상의 필드들의 반복 여부를 나타낸다. FRI와 GRI의해 반복 여부를 제한 받는 하위 필드들은 반복 코드로 구분된다. FPI 지시자의 경우, FPI 필드 다음에 연속되는 하나의 필드가 FPI의 하위 필드이다. 동일한 부모(Parent) 노드를 가지는 필드들은 트리 구조 상에서 같은 레벨에 있다고 할 수 있으며, 메시지 그 자체는 레벨 0이다. 모든 정보 필드들은 잎 노드(Leaf Node)가 된다.

단계 2) VMF 메시지 명세를 메시지 명세 세그먼트 단위로 분할

3.1절에서 정의한 ASN.1 표현 템플릿을 적용하기 위해 VMF 메시지 명세를 메시지 명세에 포함된 지시자들을 기준으로 메시지 명세 세그먼트 단위로 분할한다. 하나의 세그먼트는 부모 노드인 지시자와 지시자의 자식 노드들(지시자보다 레벨이 1만큼 큰 필드들)인 정보 필드들로 구성된다.

부모 노드인 지시자의 레벨을 각 세그먼트의 레벨로 정의한다. 정보 필드들은 모두 앞 노드가 되므로 부모 노드가 될 수 있는 것은 지시자와 루트 노드(Root Node)인 메시지 자체이다. 이 때, 루트 노드인 메시지 그 자체는 세그먼트를 구성하는 필드로 포함시키지 않고, 자식 노드들만 포함하도록 한다.

단계 3) ASN.1 표현 템플릿의 적용

단계 2에서 얻은 세그먼트들에 대해서, 3.1절에서 정의한 ASN.1 표현 템플릿을 적용한다. 이 때, 상위 레벨의 세그먼트들부터 차례로 반복적으로(Recursively) 패턴을 적용해 나간다.

레벨이 0인 세그먼트는 부모 노드로 메시지 그 자체를 가지며, 특정 지시자에 의해 구분된 세그먼트가 아닌 필드들의 단순한 나열이다. 따라서 메시지의 최상위 레벨에 해당하는 이 세그먼트는 ASN.1의 SEQUENCE 타입으로 정의한다.

세그먼트의 자식 노드가 또 다시 트리 구조를 가지는 경우, 즉 자식 노드가 지시자인 경우, 해당 지시자의 ASN.1 타입을 새롭게 선언한다. 그리고 새롭게 선언된 타입에 대해서는, 패턴들을 반복적으로 적용해 나가는 중에, 해당 자식 노드를 부모 노드로 가지는 세그먼트를 표현할 때 정의하게 된다.

4. VMF 메시지 명세의 ASN.1 표현 예제

이 장에서는 3장에서 제시한 방법을 이용하여 VMF 메시지 명세를 ASN.1 으로 표현하는 예제를 수행한다. 그림 2와 같은 VMF 메시지 명세를 생각해 볼 수 있다.

인덱스	자료항목	비트수	그룹코드	반복코드
1.	GPI	1	G1	
1.1	Year	7	G1	
1.2	Month	4	G1	
1.3	Day	5	G1	
2.	GPI	1	G2	
2.1	GRI	1	G2	R1
2.2	Type	6	G2	R1
2.3	FPI	1	G2	R1
2.3.1	Identification	4	G2	R1
2.4	Status	4	G2	R1
2.5	FRI	1	G2	R1
2.5.1	FRI	1	G2	R1/R2
2.5.2	Target number	28	G2	R1/R2
3.	FPI	1		
3.1	Comment	1400		

그림 2. VMF 메시지 명세 예제

단계 1) VMF 메시지 명세의 계층 구조 분석

메시지 자체는 레벨이 0인 루트 노드가 된다. 첫 번째 필드인 1. GPI는 레벨 1이 되고, 1. GPI와 같은 그룹 코드 G1을 가지는 세 개의 하위 필드(1.1 Year, 1.2 Month, 1.3 Day)는 1. GPI의 자식 노드가 된다. 같은 방식으로 2. GPI는 레벨 1의 노드가 되고, 2. GPI의 자식 노드로 2. GPI에 의해 그 존재 여부를 제한 받는 그룹 코드가 G2인 하위 필드들을 생각할 수 있다. 이 때, 이 하위 필드들 모두가 반복 코드 R1을 가지며, 2.1 GRI에 의해 그 반복 여부를 제한받는다. 이렇게 여러 지시자에 의해 그 존재 및 반복 여부를 제한 받는 필드들은 자신의 가장 인접한 지시자를 부모 노드로 가진다.

즉, 지시자에 뒤따르는 하위 필드들 중에 지시자가 존재하면(이를 하위 지시자라고 하자), 하위 지시자의 영향을 받는 필드들은 하위 지시자의 자식노드가 되고, 하위 지시자만이 지시자의 자식노드가 된다. 이에 따라, 2. GPI의 자식 노드는 2.1 GRI가 되고, 나머지 하위 필드들은 2.1 GRI의 자식노드가 된다. 이하 필드들에 대해서도 마찬가지로 반복적으로 진행한다. 그림 2의 메시지 명세에 대한 계층 구조를 분석한 결과는 표 2와 같다.

표 2. 단계 1-VMF 메시지 예제의 계층 구조

	Level. 1	Level. 2	Level. 3	Level. 4	Level. 5	
message	1. GPI	1.1 Year				
		1.2 Month				
		1.3 Day				
	2. GPI	2.1 GRI	2.2 Type			
				2.3 FPI	2.3.1 Identification	
			2.4 Status			
			2.5 FPI	2.5.1 FRI	2.5.2 Target number	
			3.1 Comment			
	3. FPI					

단계 2) VMF 메시지 명세를 메시지 명세 세그먼트 단위로 분할

표 2로부터 레벨이 1만큼 차이 나는 부모 자식 관계의 필드들을 하나의 세그먼트 단위로 분할해내고, 부모 노드인 지시자의 레벨을 세그먼트의 레벨로 정의한다.

메시지 그 자체는 레벨이 0이므로, 가장 상위 레벨을 가지는 세그먼트는 표 3의 SEG1로 메시지 그 자체와 레벨 1인 노드 1.GPI, 2.GPI, 3.FPI로 구성된다. 다음으로, 레벨 1에 있는 세 개의 지시자들을 기준으로 각각 세 개의 세그먼트를 분할해 낼 수 있다. 1. GPI를 부모 노드로 가지는 세그

먼트는 1. GPI와 하위 필드 1.1 Year, 1.2 Month, 1.3 Day로 구성되고, 세그먼트 레벨은 1이 된다. 이는 표 3의 SEG2에 해당한다.

나머지 노드들에 대해서도 각 레벨에 있는 지시자를 기준으로 분할을 순차적으로 진행한 결과는 표 3과 같으며, 그림 2의 메시지 명세 예제에 대해서 총 8개의 세그먼트들을 분할해내었다.

표 3. 단계 2-VMF 메시지 예제의 메시지 명세 세그먼트

세그먼트	레벨	인덱스	자료항목	비트수	그룹 코드	반복 코드
SEG1	0	1.	GPI	1	G1	
		2.	GPI	1	G2	
		3.	FPI	1		
SEG2	1	1.	GPI	1	G1	
		1.1	Year	7	G1	
		1.2	Month	4	G1	
		1.3	Day	5	G1	
SEG3	1	2.	GPI	1	G2	
		2.1	GRI	1	G2	R1
SEG4	1	3	FPI	1		
		3.1	Comment	1400		
SEG5	2	2.1	GRI	1	G2	R1
		2.2	Type	6	G2	R1
		2.3	FPI	1	G2	R1
		2.4	Status	4	G2	R1
		2.5	FPI	1	G2	R1
SEG6	3	2.3	FPI	1	G2	R1
		2.3.1	Identification	4	G2	R1
SEG7	3	2.5	FPI	1	G2	R1
		2.5.1	FRI	1	G2	R1/R2
SEG8	4	2.5.1	FRI	1	G2	R1/R2
		2.5.2	Target number	28	G2	R1/R2

단계 3) ASN.1 표현 템플릿의 적용

표 3에서 얻은 8개의 세그먼트들에 대해서, 상위 레벨의 세그먼트들부터 차례로 3.1절에서 정의한 ASN.1 표현 템플릿을 적용한다. 우선 레벨이 0인 메시지 그 자체에 대한 세그먼트 SEG1은

SEQUENCE 타입으로 정의한다. 나머지 세그먼트들은 각 세그먼트의 첫 번째 필드인 지시자를 기준으로 표 1의 템플릿을 선택하여 적용한다.

이 때, SEG1의 첫 번째 필드인 1. GPI는 지시자로 하위 노드들을 가진다. 따라서, 1. GPI에 해당되는 컴포넌트의 타입을 새로운 이름을 가지는 타입으로 선언하고, 이에 대해서는 1. GPI를 부모 노드로 가지는 SEG2를 표현할 때 타입 배정을 한다. 즉, 표 4의 세그먼트 SEG1에 대한 ASN.1 표현에서 1. GPI에 대한 컴포넌트 `idx1`을 `SEG-IDX1-GPI` 라는 새로운 타입으로 선언한다. 그리고 1. GPI를 부모 노드로 가지는 세그먼트 SEG2를 표현할 때, SEG2의 타입을 `SEG-IDX1-GPI` 타입으로 정의하고 표 1의 세그먼트-GPI 템플릿을 적용한다.

나머지 각 세그먼트에 대해서도 템플릿을 적용하여 ASN.1으로 표현해나간다. 최종적으로, 각 세그먼트에 대한 ASN.1 표현을 연결하면 주어진 VMF 메시지 명세에 대한 ASN.1 표현이 완성된다. 그림 2의 VMF 명세를 ASN.1 으로 표현한 결과는 부록 I 에 나타나있다.

5. 결론

본 논문은 VMF 메시지를 국제 표준의 정형 기법인 ASN.1을 이용하여 표현하는 체계적인 방법을 제안하였다. VMF 메시지 명세를 ASN.1으로 정의함으로써, VMF 메시지를 비트열 형태로 부터 구조적인 자료 구조로 기술할 수 있으며 VMF 메시지의 구조와 그 값을 고 수준에서 정확하고 편리하게 다룰 수 있다. 또한, 신규 메시지의 추가나 기존 메시지의 변경 시에, 소프트웨어를 새로 만들거나 수정할 필요 없이 해당 메시지 명세만

ASN.1으로 표현하여 주기만 하면 ASN.1을 지원하는 다양한 도구들을 사용할 수 있어, 보다 정확하고 신속하게 지상 전술 통신 환경에 반영할 수 있다.

본 논문에서 제안하는 방법은 이를 위해 먼저 메시지의 가변성을 이루는 메시지 명세 세그먼트들에 대해 ASN.1으로 표현하는 템플릿을 정의하였다. 이어서 앞서 정의한 템플릿을 조합하여 하나의 VMF 메시지 명세를 ASN.1으로 표현하는 방법을 제시하였다. 제안하는 방법은 메시지를 세그먼트 단위로 분할하여 각각에 대해 템플릿을 적용함으로써, VMF 메시지의 구조나 길이에 영향을 받지 않고 다양한 VMF 메시지를 템플릿의 조합으로 쉽게 표현할 수 있다.

그러나 제안하는 방법은 ASN.1을 이용하여 VMF 메시지의 구문 정보만을 표현하는데 그치고 있다. 향후 연구에서는 ASN.1 또는 ASN.1과 연동하여 사용 가능한 정형 표현 기법을 이용하여 메시지 처리 규칙과 같은 VMF 메시지의 시맨틱 정보를 표현하는 방안을 연구할 필요가 있다.

참 고 문 헌

- [1] DoD (Department of Defense), MIL-STD-6017, Variable Message Format(VMF), 2002.
- [2] Kyungmin Go, Sungwon Kang, Myungchul Kim, and Jihyun Lee, "A Systematic Test Cases Generation Approach for Testing Message Length Variability", Fourth International Conference on Software Testing, Verification and Validation (ICST 2011), March 21-25, 2011, Berlin, Germany, 2011.

- [3] Jihyun Lee, Sungwon Kang, Myungchul Kim, Changsup Keum, and Kyungmin Go, "Test Generation for VMF Tactical Data Link Messages - Coping with Message Length Variability and Semantic Message Rules", The IEEE 22nd International Symposium on Software Reliability Engineering (ISSRE 2011), Nov 29 - Dec 2, 2011 Hiroshima, Japan, 2011.
- [4] ITU-T, X.680: Information technology - Abstract Syntax Notation One(ASN.1): Specification of basic notation, 2002.
- [5] ITU International Telecommunication Union, <http://www.itu.int/ITU-T/asn1/>
- [6] Oliver Dubuisson, ASN.1 Communication between Heterogeneous Systems, OSS Nokalva, 2000.
- [7] John Larmouth, ASN.1 Complete, Open Systems Solutions, 1999.
- [8] 하수철, "ASN.1 추상구문표기법 닷원", 대전대학교, 정보통신부 정보통신 연구개발 사업 연구 결과, 2000.

저자 소개



백 하 은

2011년 한동대학교 졸업(학사).

2011년~현재 KAIST 전산학과 석사과정 재학 중



강 성 원

1982년 서울대학교 졸업(학사).

1989년 Univ. of Iowa Computer Science 졸업
(석사).

1992년 Univ. of Iowa Computer Science 졸업
(박사).

1993년~2001년 Korea Telecom Principle
Researcher.

2001년~2005년 Assistant Professor, ICU.

2005년~2009년 2월 Associate Professor, ICU.

2009년 3월~현재 Associate Professor, KAIST.

2001년~현재 Adjunct Faculty of MSE Program,
Carnegie-Mellon University, USA.



김진규

2006년 대구대학교 졸업(학사).
2006년~현재 KAIST 전산학과 박사과정 재학 중



김상수

2003년 경북대학교 졸업(석사).
2003년~현재 국방과학연구소 근무 중



김정민

2007년 숙명여자대학교 졸업(학사).
2011년~현재 KAIST 전산학과 석사과정 재학 중



권구형

2003년 고려대학교 졸업(석사).
2006년~현재 국방과학연구소 근무 중

부록 I. 단계3-VMF 메시지 예제의 ASN.1 표현

메시지 명세 예제					ASN.1 표현
SEG1					<pre> EX-MSG ::= SEQUENCE { idx1 SEG-IDX1-GPI, idx2 SEG-IDX2-GPI, idx3 SEG-IDX3-FPI } </pre>
인덱스	자료항목	비트수	그룹코드	반복 코드	
1.	GPI	1	G1		
2.	GPI	1	G2		
3.	FPI	1			
SEG2					<pre> SEG-IDX1-GPI ::= CHOICE { noPresence [0]IDX1-GPI-N, presence [1]IDX1-GPI-P } IDX1-GPI-N ::= SEQUENCE { gpi BIT STRING(0'B) } IDX1-GPI-P ::= SEQUENCE { gpi BIT STRING(1'B), g1 SEG-G1 } SEG-G1 ::= SEQUENCE { idx1-1 BIT STRING(SIZE(7)), idx1-2 BIT STRING(SIZE(4)), idx1-3 BIT STRING(SIZE(5)) } </pre>
인덱스	자료항목	비트수	그룹코드	반복 코드	
1.	GPI	1	G1		
1.1	Year	7	G1		
1.2	Month	4	G1		
1.3	Day	5	G1		
SEG3					<pre> SEG-IDX2-GPI ::= CHOICE { noPresence [0]IDX2-GPI-N, presence [1]IDX2-GPI-P } IDX2-GPI-N ::= SEQUENCE { gpi BIT STRING(0'B) } IDX2-GPI-P ::= SEQUENCE { gpi BIT STRING(1'B), g2SEG-G2 } SEG-G2 ::= SEQUENCE { idx2-1-gri SEG-IDX2-1-GRI } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
2.	GPI	1	G2		
2.1	GRI	1	G2	R1	
SEG4					<pre> SEG-IDX3-FPI ::= CHOICE { noPresence [0]IDX3-FPI-N, presence [1]IDX3-FPI-P } IDX3-FPI-N ::= SEQUENCE { fpi BIT STRING(0'B) } IDX3-FPI-P ::= SEQUENCE { fpi BIT STRING(1'B), idx3-1 BIT STRING(SIZE(1400)) } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
3	FPI	1			
3.1	Comment	1400			
SEG5					<pre> SEG-IDX2-1-GRI ::= CHOICE { noRecurrence [0]IDX2-1-GRI-N, recurrence [1]IDX2-1-GRI-P } IDX2-1-GRI-N ::= SEQUENCE { gri BIT STRING(0'B), r1 SEG-R1 } IDX2-1-GRI-P ::= SEQUENCE { gri BIT STRING(1'B), r1 SEG-R1, idx2-1-gri SEG-IDX2-1-GRI } SEG-R1 ::= SEQUENCE { idx2-2 BIT STRING(SIZE(6)), idx2-3-fpi SEG-IDX2-3-FPI, idx2-4 BIT STRING(SIZE(4)), idx2-5-fpi SEG-IDX2-5-FPI } </pre>
인덱스	자료항목	비트수	그룹코드	반복코드	
2.1	GRI	1	G2	R1	
2.2	Type	6	G2	R1	
2.3	FPI	1	G2	R1	
2.4	Status	4	G2	R1	
2.5	FPI	1	G2	R1	

메시지 명세 예제					ASN.1 표현
SEG6					SEG-IDX2-3-FPI ::= CHOICE { noPresence [0]IDX2-3-FPI-N, presence [1]IDX2-3-FPI-P } IDX2-3-FPI-N ::= SEQUENCE { fpi BIT STRING('0'B) } IDX2-3-FPI-P ::= SEQUENCE { fpi BIT STRING('1'B), idx2-3-1 BIT STRING(SIZE(4)) }
인덱스	자료항목	비트수	그룹코드	반복코드	
2.3	FPI	1	G2	R1	
2.3.1	Identification	4	G2	R1	
SEG7					SEG-IDX2-5-FPI ::= CHOICE { noPresence [0]IDX2-5-FPI-N, presence [1]IDX2-5-FPI-P } IDX2-5-FPI-N ::= SEQUENCE { fpi BIT STRING('0'B) } IDX2-5-FPI-P ::= SEQUENCE { fpi BIT STRING('1'B), idx2-5-1-fri SEG-IDX2-5-1-FRI }
인덱스	자료항목	비트수	그룹코드	반복코드	
2.5	FPI	1	G2	R1	
2.5.1	FRI	1	G2	R1 / R2	
SEG8					SEG-IDX2-5-1-FRI ::= CHOICE { noRecurrence [0]IDX2-5-1-FRI-N, recurrence [1]IDX2-5-1-FRI-P } IDX2-5-1-FRI-N ::= SEQUENCE { fri BIT STRING('0'B), idx2-5-2 BIT STRING(SIZE(28)) } IDX2-5-1-FRI-P ::= SEQUENCE { fri BIT STRING('1'B), idx2-5-2 BIT STRING(SIZE(28)), idx2-5-1-fri SEG-IDX2-1-4-1-FRI }
인덱스	자료항목	비트수	그룹코드	반복코드	
2.5.1	FRI	1	G2	R1 / R2	
2.5.2	Target number	28	G2	R1 / R2	