

버전 상호 호환 가능한 HL7 파서의 설계

이인근, 황도삼
 대구도시철도공사, 영남대학교
 inkeunlee@gmail.com, dshwang@yu.ac.kr

Design of an Version-Compatible HL7 Parser

In Keun Lee, Dosam Hwang
 Daegu Metropolitan Transit Corporation, Yeungnam University

요 약

의료정보시스템의 상호운용을 위해 개발된 의료정보 교환 국제 표준인 HL7은 복잡한 구조와 문법으로 인해 컴퓨터 소프트웨어로 관리되고 있다. 현재 개발되고 있는 HL7 인터페이스 소프트웨어에서는 다양한 버전 간 호환이 되지 않아 의료정보시스템에서 버전 상호 간의 호환을 위해 변환 소프트웨어 모듈을 개발하여 사용한다. 그러나 다양한 버전(V2.1~V2.8)의 HL7 메시지 간 상호 변환을 위해 소프트웨어 모듈을 모두 개발하는 것은 많은 시간과 막대한 비용 및 노력이 필요한 비효율적인 방법이다. 따라서 본 연구에서는 HL7 버전 호환성 정의에 기반을 두어 버전별 상호변환이 가능한 HL7 파서(Parser)를 설계하고, 객체 지향적 구조에 기반을 두어 하위 버전과의 호환(Backward Compatibility)뿐만 아니라 상위 버전과 호환(Forward Compatibility) 가능한 파서를 제안한다. 또한, 버전 간 변환 실험을 통해 효용성을 검증하였다.

주제어: HL7 V2, 버전 호환성, 객체지향 프로그래밍, HL7 인터페이스 소프트웨어

1. 서론

의료정보의 전산화를 위한 노력의 결과로서 국제적으로 전자의무기록(EMR), 평생전자건강기록(EHR) 등과 같은 의료정보의 생산과 관리를 위해 다양한 의료정보시스템이 개발됐다. 그리고 의료기관마다 독자적으로 구축되고 있는 의료정보 시스템 사이의 정보 교환을 위해 HL7[1,2], DICOM, CDA 등과 같은 다양한 의료정보 전송 표준이 사용되고 있다[3]. 특히, HL7은 이기종 시스템 사이의 의료정보 교환 목적으로 많이 사용되고 있으나, HL7 메시지는 구조와 문법이 복잡하여 컴퓨터 소프트웨어 파서(Parser)를 이용하여 처리되고 있다. HL7 메시지의 처리를 위해 Symphonia, NeoBrowse TCP, Chameleon, LINKTools, HAPI 등이 개발되었으나[2] 이들 소프트웨어는 HL7의 재사용성과 상호운용성을 보장하기 위한 버전(V2.1~V2.8) 간의 호환성이 충족되지 않아, 서로 다른 버전의 HL7 메시지를 이용하는 시스템 사이에는 버전 간 변환을 위한 새로운 모듈을 개발해야만 한다.

본 연구에서는 HL7에 정의된 버전 호환성 요구사항[4]에 따른 하위 버전과의 호환성(Backward Compatibility)뿐만 아니라, HL7 파서의 활용성을 높이기 위해 상위 버전과의 호환성(Forward Compatibility)을 보장하는 HL7 파서의 구조를 설계한다. 또한, 제안한 구조를 기반으로 ADT_A01 메시지를 분석할 수 있는 HL7 파서를 구현하고, 버전 간의 호환성 실험을 수행하여 효용성을 확인한다.

2. 관련 연구

HL7 버전 호환성 정의[4]는 (1)상위 버전(예:V2.4)의 메시지를 수신하는 하위 버전(예:V2.5)의 시스템은 오류 없이 메시지를 수신할 수 있어야 하고, (2)상위 버전의

시스템은 하위 버전의 메시지를 이해할 수 있어야 한다. 즉, HL7에서 하위 버전의 구조는 상위 버전의 구조에 포함되므로 상위 버전의 메시지를 하위 버전의 메시지로 정보 손실 없이 변환 가능하여야 한다. 그러나 실제로는 버전 간의 상이한 구조로 인해 이미 개발된 파서들에서는 버전 호환성을 충족하지 못하고 있다. 이에 [3,5]에서는 상위 버전에서 하위 버전으로의 변환이 가능한 파서가 설계되었다. 그러나 하위 버전에서 상위 버전으로의 변환을 통해 Forward Compatibility를 보장하는 파서는 전무하다.

3. 버전 상호 호환 가능한 HL7 파서의 설계

버전 간 호환 가능한 파서의 구조를 설계하기 위하여 객체지향 프로그래밍의 상속 개념을 이용하였다. 즉, 그림 1과 같이 하위 버전의 HL7 Message 클래스 객체를 상위 버전 클래스 객체가 상속하도록 함으로써, HL7 메시지의 버전에 무관하게 최하위 클래스 객체만을 사용하여 모든 버전의 HL7 메시지를 처리할 수 있도록 하였다. 예

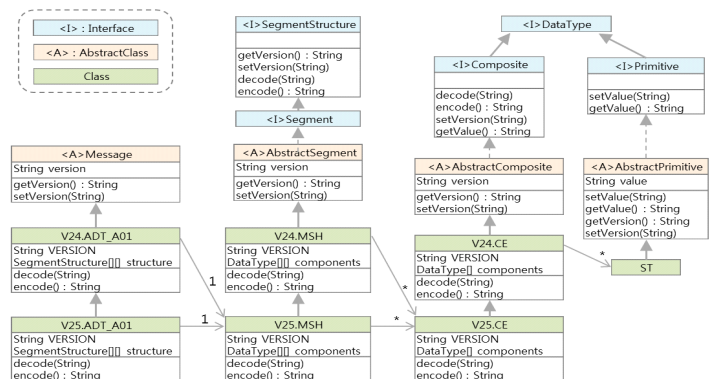


그림 1. 객체지향형 HL7 파서의 구조

를 들어, 그림 2와 같이 HL7 V2.4의 ADT^A01 메시지를 처리한다면, 파서는 메시지의 실제 버전을 먼저 확인한 후, 해당 버전의 클래스인 V24.ADT_A01 클래스 객체에서 메시지가 처리된다. V24.ADT_A01 클래스는 MSH Segment 객체를 포함하는데, 이 또한 Segment 최하위 클래스 객체인 V25.MSH에 처리를 요청하고, 실제 메시지의 버전에 따라 V24.MSH 클래스 객체에서 HL7 메시지를 전달하여 처리하도록 하였다. 이러한 과정을 통해 처리된 데이터는 해당 버전의 클래스 객체 내의 데이터로 저장된다.

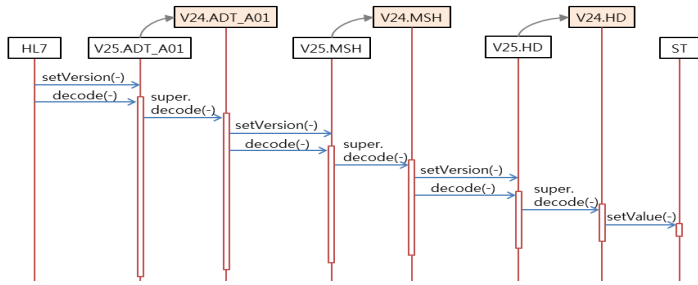


그림 2. HL7 V2.4 메시지 처리 과정

그림 3은 상위 버전(V2.5)의 메시지를 하위 버전(V2.4)으로 변환하는 과정을 보인다. 즉, HL7 V2.5 메시지를 처리한 V25.ADT_A01 객체에서 버전 설정(V2.4) 후 encode() 메소드를 호출하였으며, 버전에 따라 V25.ADT_A01의 부모클래스인 V24.ADT_A01의 encode() 메소드를 호출한다. 또한, 동일한 방법으로 V24.MSH, V24.HD의 encode() 메소드를 호출함으로써 설정된 버전의 메시지를 생성한다. 이는 하위 버전과의 호환성 (Backward Compatibility)을 위해 객체지향의 Method overriding을 이용한 방법이다.

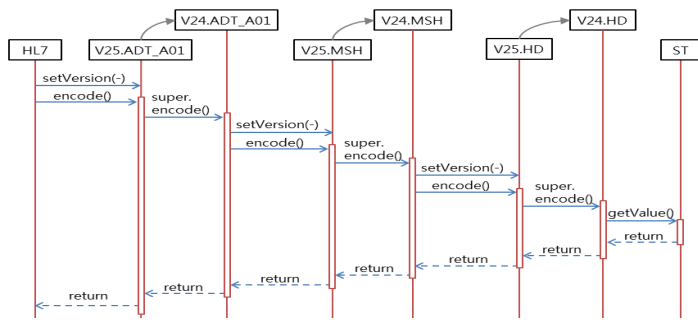


그림 3. HL7 V2.4 메시지 생성 과정

그림 4는 하위 버전(V2.4)의 메시지를 상위 버전(V2.5)으로 변환하는 과정을 보인다. 최하위 클래스인 V25.ADT_A01의 객체에서 encode() 메소드를 호출하면, HL7 V2.4 메시지를 처리한 V24.ADT_A01 객체의 encode() 메소드를 호출하고, 설정된 버전(V2.5)에 따라 V24.ADT_A01 객체 내에 저장된 데이터를 V25.ADT_A01 객체에 적합한 데이터로 변환하여 저장한다. 그리고 다시 V25.ADT_A01객체의 encode() 메소드를 호출하여 설정된 버전의 메시지를 생성한다. 이는 상위 버전과의 호환성 (Forward Compatibility)을 위해 객체지향의 Casting을 이용한 방법이다.

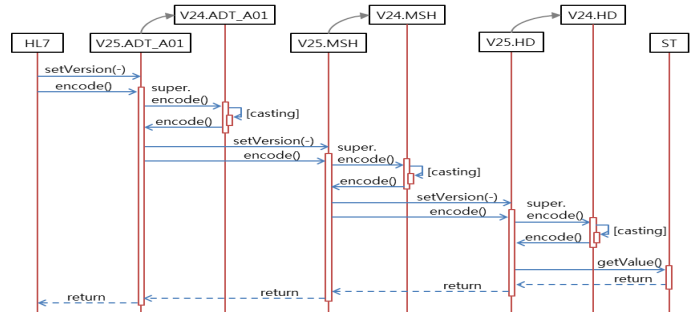


그림 4. HL7 V2.4→V25 메시지 변환 과정

제한한 설계에 따라 ADT_A01 메시지의 처리를 위한 파서를 Java 프로그래밍 언어로 구현하고, 두 가지 버전 (V2.4, V2.5)의 HL7 메시지에 대한 상호 변환 실험을 수행하였다. 다음 MSH Segment의 버전 간 상호 변환 결과에서와 같이 버전 별 Segment 규칙에 따라 메시지가 변환됨을 확인하였다.

<p>• Backward Compatibility Test (V2.5 → V2.4) V2.5 : MSH ^~\ & MegaReg UABHospC ImOrdMgr UABImgCtr 20010529090131-0500 ADT^A01^ADT_A01 01052901 P 2.5 →V2.4 : MSH ^~\ & MegaReg UABHospC ImOrdMgr UABImgCtr 20010529090131-0500 ADT^A01^ADT_A01 01052901 P 2.4</p>
<p>• Forward Compatibility Test (V2.4 → V2.5) V2.5 : MSH ^~\ & MegaReg UABHospC ImOrdMgr UABImgCtr 20010529090131-0500 ADT^A01 01052901 P 2.4 →V2.4 : MSH ^~\ & MegaReg UABHospC ImOrdMgr UABImgCtr 20010529090131-0500 ADT^A01^"" 01052901 P 2.5</p>

4. 결론

본 연구에서는 버전 상호 호환을 위한 객체 지향적 구조에 기반을 두어 HL7 파서를 설계하였고, 실험을 통해 제한한 구조의 효용성을 확인하였다. 제한한 방법을 의료정보시스템에 적용하기 위해서는 객체 내에 저장된 데이터의 입출력을 위한 메소드의 효과적인 구현에 대한 추가 연구가 필요하다.

참고문헌

- [1] HL7, Health Level Seven, Available from: <http://www.hl7.org>(Sep.2017)
- [2] HL7 Korea, Health Level Seven(HL7)과 개발도구, 한국보건산업진흥원, 2002.
- [3] 박현상, 김화선, 조훈, "호환 가능한 HL7 파서의 개발", 한국산학기술학회논문지, 제15권, 제7호, pp.4290-4300, 2014.
- [4] Health Level Seven Inc. HL7 Messaging Standard V2.5: An Application Protocol for Electronic Data Exchange in Healthcare Environments, 2003.
- [5] 박현상, 이인근, 김화선, 조훈, "이전 버전과 호환 가능한 HL7 파서의 설계", 2013 KSMI 춘계학술대회, 2013.

이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2017R1A2B4009410)