

의료정보 공유를 위한 HL7 인터페이스 엔진 구현

황 득 영*

Implementation of HL7 Interface Engine for Medical Information Exchange

Deukyong Hwang*

요 약

인터넷을 기반으로 하는 웹의 급속한 발전으로 의료 환경에 중요한 변화를 가져오고 있다. 서로 다른 병원시스템을 가진 병원들이 의료 정보를 효율적으로 공유하고 교환할 수 있는 시스템이 요구되고 있다. 시스템 간 의료정보를 전달하기 위해서는 의료 데이터를 의료정보 교환 프로토콜인 HL7에 맞춰 변환시켜주는 인터페이스 엔진이 필요하다. HL7은 의료 정보 시스템간 데이터 교환을 위한 표준 프로토콜이다. 본 논문에서는 HL7에서 정의한 메시지 구조를 바탕으로 요양기관의 치매환자를 위한 인터페이스 엔진을 구현하였다. 인터페이스 엔진은 클라이언트와 서버 모듈로 구성하였다. 클라이언트 시스템에서는 환자의 의료 데이터 입력과 메시지를 생성하여 서버 시스템으로 전달하도록 하였고, 서버 시스템에서는 메시지를 세그먼트, 필드 단위로 파싱하여 메시지의 이상 유무를 클라이언트 시스템으로 전송하도록 구현하였다. 구현 언어로는 Java와 JavaCC를 사용하였다. 본 연구로 인하여 평생전자의무기록과 원격진료시스템, 다양한 의료기관 간 의료 정보를 공유할 수 있는 기반이 될 것이다.

Abstract

Recently supply of Internet is bringing a important change in medical environments. The hospitals which had a different system is required the system that can efficiently share and exchange medical information. In order to transmission medical information between systems, the Health Level Seven(HL7) interface engine development that can convert medical data to HL7 messages is necessary. The HL7 is a standard protocol for data exchange in healthcare environments. In this paper, I implemented HL7 interface engine for Alzheimer's disease in elderly care facility. The interface engine is composed of the client system and the server system. The client system inputs user's medical care data for the aged, and builds them into HL7 message stream. HL7 messages in the client system transmitted over TCP/IP protocol to the server system. The server system parses and validates this messages stream to the segments and fields and then transmits acknowledgement to the client system. I implemented it using the Java and JavaCC. The study of interface engine implementation can be used meaningfully in electronic health record, telemedicine system, and medical information sharing among various healthcare institutions.

▶ Keyword : 헬스레벨 세븐(HL7), 노인요양시설(Elderly Care Facility), 보건의료정보시스템(Healthcare Information System)

• 제1저자 : 황득영

• 투고일 : 2010. 06. 24, 심사일 : 2010. 06. 30, 게재확정일 : 2010. 07. 22.

* 강원대학교 삼척캠퍼스 컴퓨터공학과 교수

1. 서론

최근 컴퓨터 및 정보통신 기술의 급속한 발전으로 인하여 의료 영역의 정보화도 매우 급속도로 이루어지고 있다. 보건 의료에 정보 통신 기술을 활용함에 따라 e-헬스로 통칭되는 분야도 급속한 변화가 나타나고 있다. 의료 정보화는 수많은 정보를 손쉽게 정리하여 활용할 수 있게 하는 것 이상으로 의료 시스템에 많은 변화를 가져오고 있다[1].

의료 서비스의 궁극적인 목적은 질병에 걸린 환자에게 보다 나은 진료를 통해 환자가 건강한 생활을 할 수 있도록 하는 것이다. 갑작스런 사고나 질병의 발생으로 인한 응급 상황에서는 빠르고 정확한 진료만이 환자의 생명을 구할 뿐 아니라, 완쾌후의 정상적인 생활을 보장 할 수 있다. 그러므로 응급 환자가 발생한 후 얼마나 빠른 시간 내에 정확한 진료를 하는가는 매우 중요한 사항이다. 응급환자가 작은 의원에서 진단을 받은 후에 수술이 필요하여 종합병원으로 이송되는 경우를 생각해 보면, 환자가 이송되는 동안 이루어지는 기초적인 검사 정보가 종합병원으로 전송되어 환자가 도착하기 전에 미리 환자 상태를 파악하여 필요한 수술을 미리 준비한다면 응급검사의 중복으로 인해 진료가 지연되는 것을 막을 수 있을 것이다. 이와 같은 시간적, 공간적 문제를 해결해 주는 것이 진료 정보 공유를 위한 표준화의 연구가 필요하다. 그리고 의학기술의 발달로 사람들의 평균 수명이 증가하고 물질문명이 발달함에 따라 인간의 건강에 대한 관심은 더욱 커지고 있다. 이에 따라 본인의 건강상태를 수시로 스스로 확인할 수 있고, 치료보다는 예방적인 진료 서비스가 필요해지면서 가정내 원격 진료 시스템 및 요양기관에서의 진료가 필요하며 이것도 진료 정보 공유를 위한 표준화의 연구가 필요하다[2,3].

병원 정보시스템의 급속한 증가에 비해 표준화 연구는 미흡한 실정으로 표준화되지 않은 시스템은 병원 간 정보 공유 및 교환을 어렵게 하고 향후 평생전자의무기록으로의 활용을 어렵게 한다. 서로 다른 병원 정보시스템 간에 의료데이터를 교환해야 하는 경우가 생기며, 과단위로 업무 처리하는 한 병원 내에서도 여러 회사에서 개발된 프로그램을 사용하는 경우가 흔하다. 또한 각 병원 자체 내에서의 독자적인 시스템 개발로 인하여 그 병원 내의 의사, 간호사, 검사실, 약국, 그 병원을 이용하는 환자 등으로 그 범위가 국한되어 있다. 따라서 호환성이 없는 여러 프로그램 사이의 데이터 교환이 필요하게 되며, 이때에 다른 의료기관들 간의 데이터를 올바르게 주고받기 위한 표준안으로 HL7(Health Level Seven)이 개발되었다[4].

관련 연구로 미국, 영국, 독일을 비롯한 몇몇 의료 선진국

에서는 1980년대 초반부터 병원 간 환자 정보 공유를 위한 병원 시스템이 체계적으로 시행되고 있다. HL7을 기반으로 하는 많은 엔진이 개발되었다. Orion사의 Symphonia[5], Neotool사의 NeoBrowse TCP for HL7[6], Interfaceware사의 Chameleon[7], Eversolve사의 Medi7[8] 이외에도 많은 제품들이 존재한다. 이런 제품들은 HL7을 기반으로 제작되었지만 타사 제품과의 메시지 교환에는 많은 어려움이 존재하고 평생전자의무기록과 원격진료 시스템을 구축하는 데 더 많은 돈과 노력이 소요된다. 우리나라에서는 HL7이 소개된 90년대 초반 이후 10년 가까운 기간 동안 의료분야로부터 거의 관심을 끌지 못하고 있다가 21세기에 접어들면서 사회적인 정보화의 영향을 받아 의료분야 역시 정보에 대한 교환 수요가 나타남에 따라 차츰 관심이 높아져가는 추세에 있다. 이러한 추세를 보임에 따라 HL7에 대하여 관심과 흥미를 갖는 관련분야 종사자들이 늘어나고 있다. 하지만 현재 우리나라에서 시판되고 있는 상업용 HL7 관련 소프트웨어는 4~5종이 있으나 전부 외국산으로서 HL7 엔진의 수입 비용과 연간 유지보수 비용의 문제가 발생하고 있다. 또한 다른 나라에서 개발한 시스템을 국내 시스템에 맞도록 수정함이 없이 그대로 적용하기에는 제한이 있다. 즉, HL7을 기반으로 제작된 엔진들이 메시지를 교환할 때 각각의 엔진에 맞게 형식을 맞추기 위해 별도의 모듈이 존재해야 한다[9,10,11].

이러한 문제로 본 연구는 의료 시장개방에 대처하는데 어느 정도 도움이 되기 위해 HL7을 기반으로 하여 시스템 간 의료정보 메시지를 송수신하는 시스템을 개발함으로써 보건 의료정보화 촉진 및 국민 의료서비스 향상을 위해서 의료기관 간 진료정보 공동 활용의 기반을 마련하고자 한다.

본 논문에서는 HL7 표준안에 정의된 프로토콜에 따라 의료 데이터를 HL7 형식에 맞게 생성하는 메시지 생성기와 메시지 파서를 개발하였다. 의료 데이터를 교환하기 위해 HL7을 기반으로 하여 병원과 요양기관 치매환자의 데이터를 전송하기 위한 인터페이스 엔진을 구현하였다. 의료정보 교환을 위해서 요양기관에서 진료에 필요한 환자 정보와 검사 정보를 입력하게 되면 그 내용을 HL7의 메시지로 변환하여 전송하고, 메시지를 받은 병원 시스템에서는 메시지를 파싱, 검증하도록 설계하였다.

본 연구에 의하여 각 의료 기관이 상이한 의료 정보 시스템을 운용하더라도 진료 정보 교환이 가능한 의료 정보 공유 시스템을 구축할 수 있다. 고령화 사회에 따른 장기요양기관의 만성 질환자의 지속적인 진료 및 기록 관리가 가능하다. 응급 상황 발생시 원활하고 신속한 조치를 위한 의무 기록 참조가 이루어질 수 있다.

본 논문의 구성은 II장에서는 HL7의 구조에 대해 설명하고, III장에서는 시스템 설계를 위한 모델링과 HL7 메시지 설계에 대해 설명한다. IV장에서는 HL7 인터페이스 엔진을 구현하고, V장에서는 결론에 대하여 언급하고자 한다.

II. HL7 개념

HL7이란 보건의료정보시스템간 데이터 교환을 위한 표준 프로토콜이다. 1987년에 처음 개발되었으며, 현재 북아메리카에서는 의료정보의 전자적 교환을 위한 표준이다. HL7 메시지는 세그먼트, 필드, 컴포넌트로 구성되어 있다[12].

2.1 메시지

메시지는 데이터 전송의 가장 작은 단위이며, 세그먼트들로 구성된다. 메시지는 메시지 헤더 세그먼트(MSH)로 시작하고, 메시지 타입과 트리거 이벤트로 메시지와 메시지를 구분한다. 그림 1은 메시지의 구조를 보여주고 있으며, 메시지는 세그먼트들로 구성되고, 세그먼트는 필드들로 구성되어 있음을 알 수 있다.

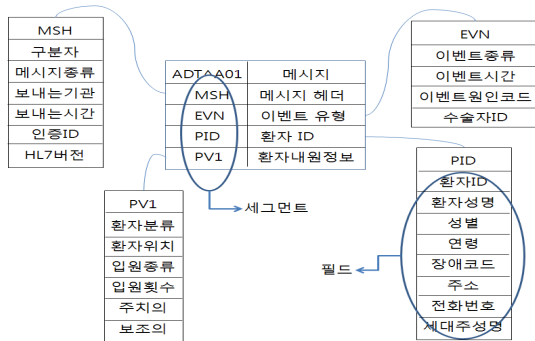


그림 1. 메시지 구조
Fig. 1. Message Structure

2.2 세그먼트

세그먼트는 정해진 순서가 있는 필드들의 논리적 집합이다. 필수 세그먼트, 선택적 세그먼트, 하위호환을 위해 남겨둔 세그먼트가 있다. 필수 세그먼트는 메시지 내에서 반드시 기입해야 하는 세그먼트이다. 선택적 세그먼트는 사용하지 않거나 여러 번 반복을 허용한다. 선택적인 세그먼트는 {segmentID}, 반복적인 세그먼트는 {segmentID}로 표시한다. 세그먼트는 세그먼트 ID라고 하는 처음 세 글자에 의해 식별할 수 있다. 세그먼트는 세그먼트 구분자(보통 <CR>)에 의해 나누어진다. 그림 1에서 메시지 헤더를 나타내는 MSH, 이벤트에 대해 기

술한 EVN, 환자의 ID에 대해 기술한 PID, 환자 방문 정보인 PV1과 같은 세그먼트들로 구성되어 있다. 그 뒤를 추가 정보가 따라붙는 형태를 갖고 있다. 예를 들어 검사 결과, 성능, 견적서 등의 정보가 있다.

2.3 필드

메시지의 실제 내용은 세그먼트 안의 필드에서 표현된다. 필드는 컴포넌트들의 집합이며, 컴포넌트는 다시 서브컴포넌트로 구성된다. 필드는 문자, 숫자, 날짜, 시간, 복합문 등 여러 가지 자료형으로 정의되고, 전송될 때는 스트링의 형태로 전송된다. 필드의 최대 길이 속성은 한 필드에 표현할 수 있는 최대 문자 개수를 정의한다. 필드는 널 값을 허용한다. 필드는 반복을 허용한다.

2.4 메시지 구분자

메시지를 구성할 때 사용되는 특수문자로서, 세그먼트 구분자, 필드 구분자, 컴포넌트 구분자, 서브컴포넌트 구분자 그리고 탈출(escape)문자가 있다. 표 1은 여러 가지 구분자들을 용도에 따라 자세히 설명해 놓은 것이다.

표 1. 메시지 구분자
Table 1. Message Delimiter

| 구분자의 종류 | 값 | 용도 |
|------------|------|---|
| 세그먼트 구분자 | <CR> | 세그먼트의 종료를 알림 이 값은 임의로 변경할 수 없음 |
| 필드 구분자 | | 세그먼트 내의 인접한 두 필드를 구분할 때 사용됨 각 세그먼트의 첫 번째 데이터 필드와 세그먼트 ID를 구분할 때도 사용함 |
| 컴포넌트 구분자 | ^ | 데이터 필드의 인접한 컴포넌트를 구분할 때 사용됨 |
| 서브컴포넌트 구분자 | & | 데이터 필드의 인접한 서브컴포넌트를 구분할 때 사용됨 |
| 반복 구분자 | ~ | 하나의 필드에서 같은 내용을 반복할 때 사용됨 |
| 탈출문자 | \ | ST, TX, FT자료형으로 표현되는 필드 컴포넌트에서 사용됨 |

III. HL7 인터페이스 모델링

트리거 이벤트에 의해 하나의 사건이 발생되면, 그 사건에 의해 데이터의 교환이 일어나게 될 두 개 이상의 시스템이 네트워크를 통해 경로가 설정되어진다. 시스템 설계를 위한 모델링은 요양기관에 있는 치매환자를 등록하고 수시로 건강상태를 파악하여 검사결과를 클라이언트 시스템에서 HL7 메시

지로 생성하여 병원 서버 시스템으로 전달하고, 메시지를 받은 병원 서버 시스템에서는 메시지를 처리하고 나서 ACK 메시지로 응답한다. ACK 메시지에 HL7 메시지를 성공적으로 전송 받았는지, 전송받은 HL7 메시지에 이상이 없는지에 대한 정보가 들어 있다[1,13].

HL7 메시지의 송수신 메카니즘은 송신측 시스템이 수신측 시스템으로부터 수신확인 메시지를 받음으로써 완전한 사이클이 이루어진다. 예를 들어 '환자입원' 이벤트가 발생하면 환자관리 시스템에서는 다른 하부시스템으로 메시지를 송신하게 된다. 이때 메시지를 수신하는 시스템에서는 '환자입원-확인'이라는 수신확인 메시지를 송신측 시스템(환자관리 시스템)에 전송함으로써 사이클을 종료한다. HL7 인터페이스 모델링은 그림 2와 같다.

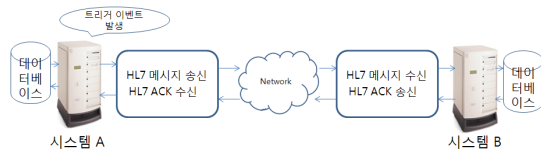


그림 2. 시스템 모델링
Fig. 2. System Modeling

3.1 환자 등록 정보 메시지 설계

표 2의 ADT^A04 추상 메시지 구문의 세그먼트들은 여러 가지 정보를 포함하고 있으나 본 논문에서는 다음과 같은 항목을 선택하여 그림 3과 같이 구성하였다.

표 2. 추상 메시지 구문
Table 2. Abstract Message Syntax

| ADT^A04 | ADT message |
|-----------|--------------------------------------|
| MSH | 메시지 헤더 |
| EVN | 이벤트 유형 |
| PID | 환자 ID(Patient Identification) |
| (PD1) | 환자정보 - 추가사항 |
| {{(ROL)}} | 병원 관련자의 기능, 책임 |
| {{(NK1)}} | 보호자정보 |
| PV1 | 환자내원정보 |
| (PV2) | 환자내원정보 - 추가사항 |
| {{(ROL)}} | 병원 관련자의 기능, 책임 |
| {{(DB1)}} | 장애정보 |
| {{(OBX)}} | 의사조견서, 결과정보 |
| {{(AL1)}} | 알려지정보 |
| {{(DG1)}} | 진단정보(Diagnosis Information) |
| (DRG) | 유사진단군 의료비지불(Diagnosis Related Group) |
| {{ PR1 | 처치정보 |
| {{(ROL)}} | 병원 관련자의 기능, 책임 |
| {{(GT1)}} | 진료비지불 - 보증인정보 |
| {{ IN1 | 보험정보 |

| | |
|-----------|-------------------|
| {(IN2)} | 보험 - 추가정보 |
| {{(IN3)} | 보험 - 추가정보 - 보험 증명 |
| {{(ROL)}} | 병원 관련자의 기능, 책임 |
| (ACC) | 사고정보 |
| (UB1) | 진료비청구서 정보 |
| (UB2) | 진료비청구서 - 부가정보 |
| (PDA) | 현자 사망, 부검 정보 |

그림 3은 이벤트 A04의 처리 구조로 HL7 메시지의 세그먼트 순서를 알 수 있다. MSH는 전송 기관, 수신 기관, 메시지 전송 날짜, HL7 버전 등을 전송하기 위한 세그먼트이고, EVN은 이벤트를 나타내는 세그먼트이고, PID는 환자에 대한 정보(ID, 이름, 주민등록번호, 생년월일, 성별, 학력, 결혼상태, 전화번호, 주소, 메일)를 위한 세그먼트이고, NK1은 가족구성원을 나타내는 세그먼트인데 여기선 자녀가 몇 명이나에 따라 생각할 수도 있고 가족 인원만큼 반복될 수도 있게 하였다. PV1 세그먼트는 외래진료라는 것과 진료담당의사를 나타내고 있다.

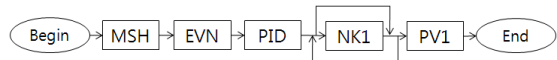


그림 3. HL7 메시지의 이벤트 A04 처리 구조
Fig. 3. The Event A04 Diagram of HL7 Message

```
MSH|^~W&|REGADT|MCM|IFENG||199112311501||ADT^A04|000001|P|2.4|||<br>
EVN|A04|199901101500|199901101400|01||199901101410<br>
PID|||191919^^^GENHOS^MR~371-66-9256|MAS SIE^JAMES^A||19560129|M|.....<br>
NK1|1|MASSIE^ELLEN|SPOUSE|171 ZOBERLEIN|.....<br>
NK1|2|MASSIE^MARYLOU|MOTHER|300 ZOBER^^ISHPEMING^MI^49849^""^|||<br>
PV1|||O|R/|||0148^ADDISON,JAMES|0148^ADDISON,JAMES|.....<br>
```

그림 4. 환자 등록 결과 메시지 예
Fig. 4. Example of Admitted Patient Result Message

그림 4는 환자 등록 결과 HL7 메시지 예로서 EVN 세그먼트 첫 번째 필드에서 이벤트 A04임을 알 수 있고, 자녀가 두 명이므로 NK1 세그먼트가 두 번 나타나는 것을 볼 수 있다. PID 세그먼트는 환자의 정보를 알 수 있고, PV1 세그먼트에서는 진료형태와 진료담당의사를 알 수 있다.

표 3. ACK^A04 추상 메시지 구문
Table 3. Abstract Message Syntax of ACK^A04

| ACK^A04 | 일반적 확인(General Acknowledgment) |
|---------|--------------------------------|
| MSH | 메시지 헤더 |
| MSA | 메시지 확인 |
| (ERR) | 오류 |

표 3은 ADT^A04 메시지에 대한 수신 확인 메시지 구문으로 각 세그먼트에 대한 설명을 볼 수 있다. ADT^A04 메시지를 수신한 시스템은 수신확인 사실을 알리기 위하여 트리거 이벤트 A04에 대한 확인 메시지로 응답, 오류, 거부를 보내기 위해 ACK와 조합한 ACK^A04을 되돌려 준다.

3.2 검사 정보 입력 메시지 설계

표 4의 세그먼트들은 여러 가지 정보를 포함하고 있으나 본 논문에서는 그림 4와 같은 항목을 선택하여 사용하였다. 그림 4는 HL7 메시지의 이벤트 R01의 처리 구조로 HL7 메시지의 세그먼트 순서를 알 수 있다.

표 4. ORU^R01 추상 메시지 구문
Table 4. Abstract Message Syntax of ORU^R01

| ORU^R01 | ORU message |
|-----------|-------------------|
| MSH | 메시지 헤더 |
| {{PID}} | 환자 ID |
| {PD1} | 환자정보 - 추가사항 |
| {{NK1}} | 보호자정보 |
| {{NTE}} | 주석과 설명 |
| {PV1} | 환자내원정보 |
| {PV2}} | 환자내원정보 - 추가사항 |
| {{ORC}} | Order common |
| OBR | 처방 및 임상 결과에 대한 기록 |
| {{NTE}} | 주석과 설명 |
| {CTD} | Contact Data |
| {{OBX}} | 의사소견서/결과정보 |
| {{NTE}}} | 주석과 설명 |
| {{FT1}} | 회계 |
| {{CT1}}}} | 임상실험 |
| {DSC} | 연속 포인터 |

ADT^A04 추상 메시지와 같이 MSH는 전송 기관, 수신 기관, 메시지 전송 날짜, HL7 버전 등을 전송하기 위한 세그먼트이고, PID는 환자에 대한 정보(ID, 이름)를 위한 세그먼트이다. OBR 세그먼트는 측정된 날짜를 전송하기 위한 것이고, OBX는 치매 환자가 측정한 검사 결과를 전달하기 위한

세그먼트이고 검사 결과가 여러 개일 경우는 반복적으로 나올 수 있다는 것도 알 수 있다.

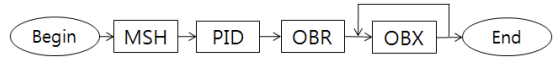


그림 4. HL7 메시지의 이벤트 R01 처리 구조
Fig. 4. The Event R01 Diagram of HL7 Message

```

MSH|^~W&|XRAY||CDB|200006021411||ORU^R01|K172|P|.....<cr>
PID||191919^^^.....|371-66-9256||<cr>
OBR|1|X89-1501^OE|78912^RD|71020^CHEST XRAY AP WTWLATERAL|||.....<cr>
OBX|1|CE|71020&IMP^RADIOLOG|4|^MASS LEFT LOWER LOBE|||A|||F|.....<cr>
OBX|2|CE|71020&IMP|2|^INFILTRATE RIGHT LOWER LOBE|||A|||F|<cr>
OBX|3|CE|71020&IMP|3|^HEART SIZE NORMAL|||N|||F|.....<cr>
OBX|4|FT|71020&GDT|1|circular density is seen in the posterior segment of L|||F|.....<cr>
OBX|5|CE|71020&REC|5|71020^Follow up CXR 1 month||30-45|||F|.....<cr>
    
```

그림 5. 검사 정보 결과 메시지 예
Fig. 5. Example of Checkup Information Result Message

그림 5는 그림 4의 구조에 맞게 환자 검사 정보를 입력했을 때의 HL7 메시지로 검사 항목에 따라 OBX 세그먼트가 반복되고 있음을 알 수 있다.

IV. HL7 인터페이스 엔진 구현

4.1 구현 환경

본 논문은 시스템 구현을 위해 HL7 version 2.4를 사용하여 클라이언트 시스템과 병원 서버 시스템간의 정보 교환이 이루어진다. 시스템 사용자 인터페이스와 HL7 서버 엔진을 위한 운영체제는 Windows 2000/XP/7를 비롯하여 모든 계열의 운영체제에서 사용이 가능하다. 구현 언어로는 자바를 사용하였으며 썬 마이크로 시스템의 jdk1.6.0를 기반으로 하였다. 메시지 검증을 위해 Java 컴파일러 생성기인 JavaCC (Java Compiler Compiler)를 이용하였다[14,15].

4.2 HL7 엔진 개발

클라이언트 시스템에서 발생하는 환자 등록과 검사 정보를 병원 서버 시스템으로 전달하기 위해 정보를 표준방식인 HL7 인코딩 규칙에 맞추어 변환하는 사용자 HL7 엔진을 개발하였다. 또한 클라이언트 시스템으로부터 인코딩 규칙에 맞추어 변환된 메시지를 받은 병원 서버 시스템은 그 메시지를 성공적으로 받았는지에 대한 메시지를 보내는 데 필요한 엔진도 개발하였다. HL7 엔진에서 일어나는 메시지 송수신 과정은 그림 6과 같다.

사용자가 환자 등록과 검사 정보 메시지를 보낼 때 사용자 HL7 엔진에서는 HL7 프로토콜에 의한 메시지 생성이 일어나며 생성된 메시지는 TCP/IP 프로토콜을 이용하여 병원 시스템 엔진으로 전송한다. 전송된 HL7 메시지는 무결성을 검증하여 디코딩하고 성공적으로 받았는지에 대한 메시지를 다시 사용자 HL7 엔진으로 보낸다.

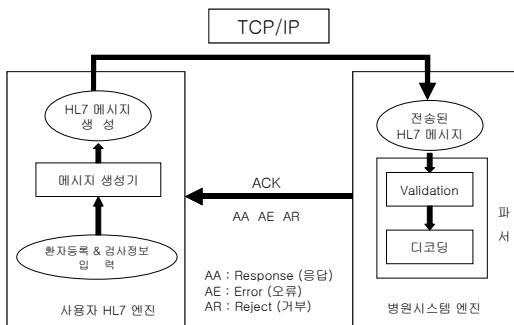


그림 6. HL7 엔진에서 일어나는 메시지 송수신 과정
Fig. 6. Message Transmission/Receiver Process

4.2.1 HL7 메시지 생성기

기본 의료정보들은 의료 정보시스템간의 교환을 위해 HL7 메시지로 변환되는 인코딩 과정이 필요하다. HL7 표준 2.4버전에서는 ASCII 텍스트 형태의 메시지로 의료 시스템 간 정보를 교환한다.

생성기는 기본 데이터를 이용하여 전송하기 위한 HL7 메시지로 바꾸어준다. 생성기에는 메시지 구조와 세그먼트 정보가 들어있어 HL7 표준의 모든 메시지별 추상 메시지 구문을 HL7 메시지 생성규칙에 따라 변환하여 표현해준다.

그림 7은 ADT_A04 메시지를 생성하기 위한 클래스로 각 세그먼트(MSH, EVN, PID, NK1, PV1)들을 각각 메소드로 호출하여 생성하도록 하였다. 또한 모든 세그먼트들은 필드와 컴포넌트 생성을 위한 부분과 생성된 결과를 반환하는 메소드들로 구성되어 있다.

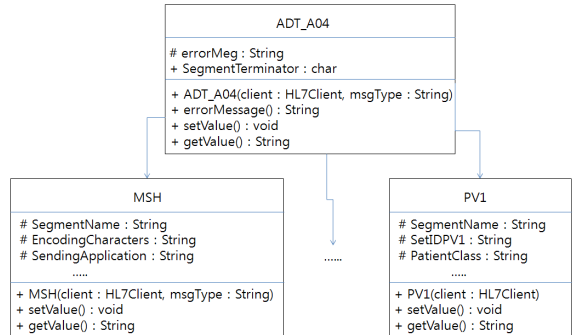


그림 7. 메시지 생성 클래스 구조
Fig. 7. Class Hierarchy of Message Generation

4.2.2 HL7 메시지 파서

파서는 HL7 2.x대의 메시지를 처리하는 다른 의료 정보 시스템에서 메시지를 받았을 경우 자신의 시스템에서 처리하기 위해 텍스트에서 필드, 컴포넌트, 서브컴포넌트의 정보를 추출해 주는 역할을 한다.

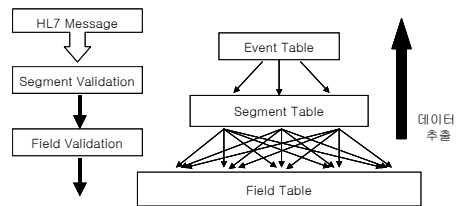


그림 8. HL7 메시지 파서 기능
Fig. 8. HL7 Message Parser Function

그림 8은 HL7 메시지를 세그먼트 단위로 분리하고 다시 필드 단위로 구분하여 각각의 세부 항목에 따라 이벤트 테이블과 세그먼트 테이블, 필드 테이블을 차례로 검색하여 해당하는 데이터를 추출한다. 하나의 이벤트에 포함되어 있는 세그먼트와 그 하부의 필드들은 이벤트 항목에 명시된 존재 여부와 반복 형태에 따라 순차적으로 파악되며, 이들 세부적 항목에 대한 검증 작업도 동시에 수행하게 된다.

검증은 생성된 메시지 또는 전송받은 메시지의 검증을 위해 Java언어 컴파일러 툴인 JavaCC를 이용한다.

그림 8에서 세그먼트 검증 과정에서는 전송되는 메시지 세그먼트 중 반드시 있어야 되는데 빠진 세그먼트나 순서가 맞지 않는 세그먼트를 검사한다.

그림 8에서 필드 검증 과정에서는 요구 필드(Require field), 필드 길이, 자료형, 컴포넌트와 서브컴포넌트를 검사한다. 요구 필드는 세그먼트 안의 필드의 선택성에 따라 필수,

선택, 비사용으로 구분된다. 반드시 있어야 하는 필드인데도 필드의 값이 없거나 필드 자체가 없는지 검사한다. 필드 길이는 HL7 표준에서는 각 필드의 최대 길이를 정의하고 있다. 표준에서 정의된 필드 길이보다 길지 않은지 검사한다. 자료형은 HL7 표준에서는 55개의 데이터 형식을 가지고 있다. 각 필드마다 정해진 데이터 형식이 있으며 이 형식을 만족하는지 검사한다. 데이터 값은 어떤 필드는 정해진 테이블에 있는 값만을 가지도록 정의되어 있다. 예를 들어 성별 코드로는 M(Male), F(Female), U(Unknown)만 올 수 있다. 컴포넌트와 서브컴포넌트는 하나의 필드 안에는 컴포넌트가 여러 개 올 수 있고, 하나의 컴포넌트 안에는 여러 개의 서브컴포넌트가 올 수 있다. 각각의 컴포넌트나 서브컴포넌트도 필드 처럼 데이터 형식, 값, 길이 등을 가지므로 컴포넌트와 서브컴포넌트에 대한 검사도 필요하다.

그림 9는 ADT_A01 메시지에 대한 JavaCC 검증 프로그램으로 메시지를 MSH와 EVN 세그먼트 단위로 분리하고 다시 필드 단위로 구분하여 데이터를 추출한다. ADT_A04와 ORU_R01도 같은 방법으로 처리된다.

```
void ADT_A01():
{
    MSH seg() EVNseg() PIDseg()
    [ LOOKAHEAD(PIDseg()) PIDseg() ]
    [ ROLsegList() ] [ NK1segList() ] PV1seg()
    [ LOOKAHEAD(PV1seg()) PV1seg() ] [ ROLsegList() ]
    [ DB1segList() ] [ OBXsegList() ] [ AL1segList() ]
    [ DG1segList() ] [ DRGseg() ]
    [ PR1segROLsegListList() ]
    [ GT1segList() ]
    [ IN1segIN2OptSegIN3segListROLsegListList() ]
    [ ACCseg() ] [ UB1seg() ] [ UB2seg() ]
    [ PDAseg() ]
}
```

그림 9. ADT_A01 메시지에 대한 JavaCC 검증
Fig. 9. JavaCC validation of ADT_A01 Message

모든 검증이 끝나고 난 뒤 수신 시스템은 전송 시스템에 해당 메시지에 대한 처리 결과를 통보하게 된다. 이때 전달되는 메시지는 응답, 오류, 거부 세 가지 중에서 하나가 된다.

4.3 시스템 구현 결과

요양기관 치매 환자 정보를 HL7 메시지로 교환하기 위해서는 우선 통신 설정이 필요하다. 그림 10의 인터페이스 엔진

에서 통신 설정 화면 탭을 클릭하여 "Connect" 버튼을 클릭하면 통신 설정이 된다. 본 논문에서는 메시지의 송신 및 수신 기능으로서 TCP/IP 프로토콜을 이용하여 메시지의 송수신을 가능하게 하였다.

그림 10은 환자 등록 화면을 보여주는 예로 병원에서 부여 받은 ID와 환자 신상정보를 입력한 것이다. 모든 항목을 입력하고 "등록" 버튼을 클릭하면 병원으로 환자 등록 메시지가 보내진다. "등록" 버튼을 클릭하여 오류가 없을 때에는 그림 11과 같은 확인창이 열린다. 오류가 있다면 어느 항목이 잘못 입력되었는지 알려주는 새로운 창이 열린다.

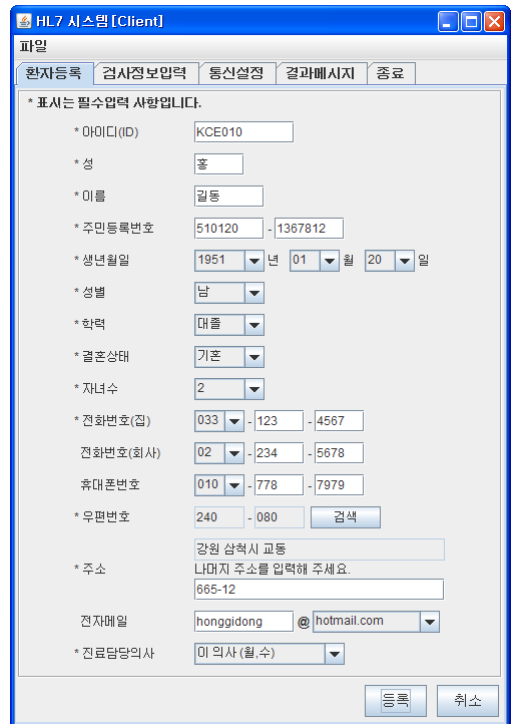


그림 10. 환자 등록 화면
Fig. 10. Patient Register Screen

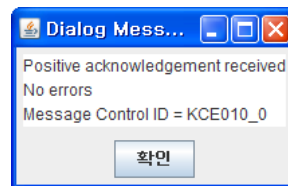


그림 11. 확인 화면
Fig. 11. Acknowledge Screen

그림 12는 환자 등록을 했을 때 결과 메시지 화면으로 병

원 서버 시스템으로 보낸 HL7 메시지와 받은 HL7 메시지를 볼 수 있다.

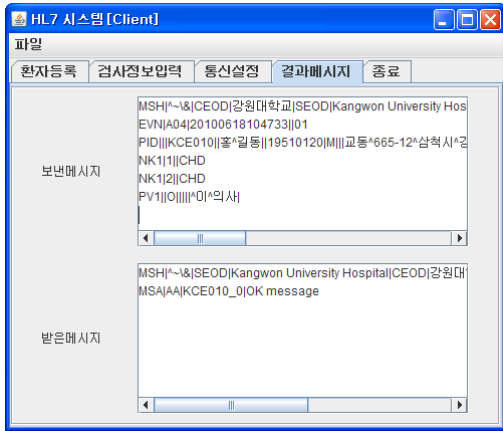


그림 12. 환자 등록 결과 메시지 화면
Fig. 12. Admitted Patient Screen

그림 13은 검사 정보 입력 화면으로 ID를 입력하고 "확인" 버튼을 클릭하면 병원 시스템의 데이터베이스에서 가져온 환자의 성, 이름, 주민등록번호가 나타나게 된다. 검사할 때마다 환자 등록을 해야 하는 불편함을 없앨 수 있다.

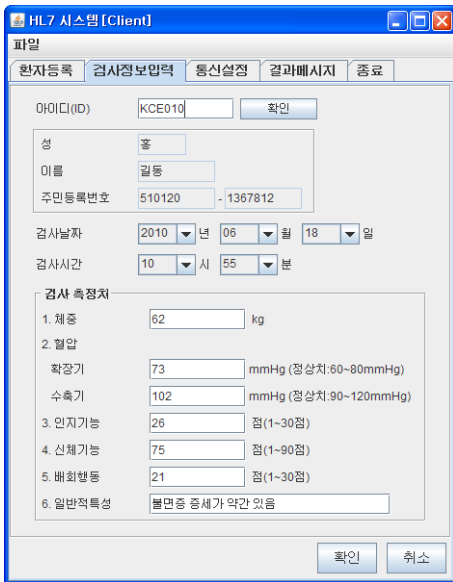


그림 13. 검사 정보 입력 화면
Fig. 13. Checkup Information Input Screen

그림 13은 검사날짜, 검사시간, 검사 측정치를 입력한 화

면이다. 치매 환자를 위한 검사측정치는 인지기능, 신체기능, 배회행동을 검사하도록 하였다[16,17]. 인지기능은 지남력, 주의집중, 언어기능, 판단력 등의 평가도구를 포함하고 있다. 신체기능은 일상생활활동 측정, 신체적 일상생활 활동, 도구적 일상생활 활동 등의 평가도구를 포함하고 있다. 배회행동은 걸음, 보행형태, 이탈행동, 공간적 방향상실 등의 평가도구를 포함하고 있다. "확인" 버튼을 클릭하면 병원으로 검사 정보 메시지가 보내진다. 오류가 없다면 그림 11과 같은 창이 열린다. 오류가 있다면 어느 항목이 잘못 입력되었는지 알려 주는 새로운 창이 열린다. 검사날짜와 검사시간은 현재 날짜와 시간이 나타나게 하였다.

그림 14는 환자가 검사 정보를 입력했을 때의 결과 메시지 화면으로 병원 서버 시스템으로 보낸 HL7 메시지와 받은 HL7 메시지를 볼 수 있다.

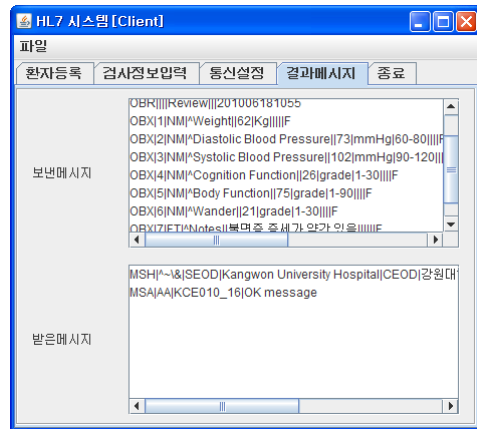


그림 14. 검사 정보 입력 결과 메시지 화면
Fig. 14. Checkup Information Result Screen

그림 15는 사용자가 접속했을 때의 병원 서버 시스템 화면으로 Incoming Message를 보면 ADT^A04와 ORU^R01 메시지를 받은 날짜와 시간, 실제 HL7 메시지를 볼 수 있다. Outgoing Message에서는 두 메시지를 오류없이 잘 받았다는 HL7 메시지를 되돌려 주었음을 알 수 있다.

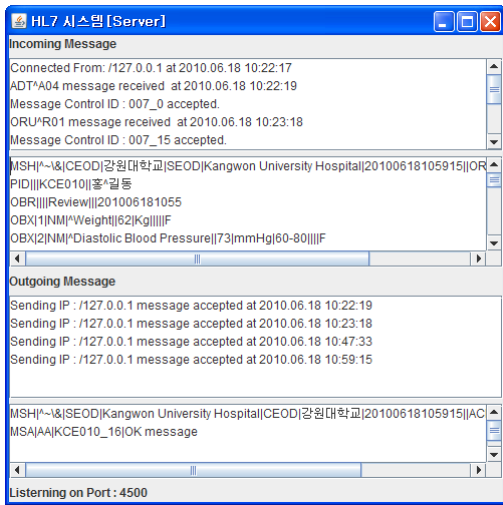


그림 15. 병원 서버 시스템 화면
Fig. 15. Hospital Server System Screen

병원시스템의 서버는 여러 사용자(환자)들과 연결되어 있는데 그 사용자들을 구분하기 위해 접속/종료된 날짜와 시간, IP주소가 각각 기록된다. 또한 병원시스템 서버로 들어온 메시지에 대한 수신 확인 메시지의 날짜와 시간, IP주소도 기록된다. 사용자는 건강 상태에 따라 수시로 병원시스템의 서버에 접속할 수 있으므로 접속되어 있는 사용자가 없다고 해서 병원 서버 시스템을 종료해서는 안된다.

4.4 시스템 평가

시스템 평가 방법은 현재 상용으로 사용 중인 시스템과의 HL7 메시지 전송을 위한 호환성을 검사하였다. 현재 시판되고 있는 HL7 메시지 생성 툴인 Symphonia(5), NeoBrowse(6), Chameleon(7), Medi7(8) 엔진들과 본 엔진 간의 호환성을 확인해 본 결과 오류 없이 의료정보가 전송됨을 확인하였다. 타 시스템들은 서로 다른 엔진 간에 메시지를 교환할 때 형식을 맞추기 위해 별도의 모듈이 존재하지만 본 시스템은 필요 없다. 본 연구로 구현된 HL7 인터페이스 엔진이 타사 제품과 호환을 이룸으로써 서로 다른 병원과도 의료정보 공유 및 교환이 가능하다.

V. 결 론

의료분야의 정보화가 활성화됨에 따라 서로 다른 병원 정보시스템 간에 의료정보를 교환해야하는 경우가 생기며, 호환성이 없는 여러 프로그램 사이의 정보교환이 필요하게 되는데

이때 데이터를 올바르게 주고받기 위한 표준안으로 HL7이 개발되었다. 진료 기관이 각기 다른 의료정보 시스템을 운용하면 전자적 진료 정보교환이 어렵기 때문에 진료 문서 공유 표준화를 준수한 새로운 시스템이 필요하다. 진료의 속성상 환자 정보는 다수의 의료 기관에 산재하여 진료에 필요한 여러 가지 다양한 정보가 워낙 많은 의료 정보 시스템에 독자적인 데이터 관리 형태로 저장되어 있기 때문에 하나의 독자적인 시스템으로의 통합 또는 표준화는 불가능하다, 그래서 각기 의미 있게 활용하는 독자적인 의료 정보 시스템을 유지하는 한편 최소한의 진료 정보 공유 체계에 근거한 인터페이스 표준화 방안이 절실히 필요하다.

본 논문은 의료정보 교환 표준 프로토콜인 HL7을 기반으로 하여 요양기관의 치매 환자와 병원 시스템과 의료정보 교환을 위한 인터페이스 엔진을 구현하였다.

본 논문은 클라이언트 시스템에서 발생하는 환자 정보와 검사 정보를 병원 서버 시스템으로 보내기 위해 HL7 인코딩 규칙에 맞추어 HL7 메시지를 자동으로 생성하여 서버로 전송하는 인터페이스 엔진을 개발하였다. 그리고 클라이언트 시스템으로부터 변환된 메시지를 받은 병원 서버 시스템이 그 메시지를 성공적으로 받았는지에 대한 수신확인 메시지를 보내는 데 필요한 엔진도 개발하였다. 병원 서버 시스템에서는 전송받은 메시지의 검증을 위해 Java언어 컴파일러 툴인 JavaCC를 이용하여 세그먼트와 필드, 컴포넌트를 검증하여 메시지가 HL7 규칙에 맞는지를 검사하도록 하였다.

본 연구로 인하여 병원 자체 내에서만 이루어지던 진료 서비스가 병원과 병원, 병원과 약국, 검사실과 수술실, 병원과 환자 등과 같이 그 범위가 확대될 수 있다. 그로 인해 환자가 병원을 옮길 때마다 중복된 검사를 해야 하는 시간적, 금전적 낭비도 어느 정도 해결할 수 있게 될 것이다. 더 나아가 HL7을 기반으로 하는 다른 원격 진료 서비스도 쉽게 개발할 수 있다. 응급 상황 발생시 원활하고 신속한 조치를 위한 의무 기록 참조가 이루어질 수 있다. 고령화 사회에 따른 장기요양기관의 만성 질환자의 지속적인 진료 및 기록 관리가 가능하다.

이번 연구에서는 HL7 인터페이스 엔진 구현에 중점을 두었으나 앞으로 병원 서버 시스템의 임상 데이터베이스를 HL7 표준에 맞도록 구축하여 환자 의료 정보를 통합하여 국민 평생 진료 기록 공유가 가능하도록 개발하는 것이 필요하다.

참고문헌

[1] 정수희, "임산부 의료정보 교환을 위한 HL7 인터페이스 엔진 구현," 강원대학교, 석사학위논문, 2005.

- [2] 김정원, "모바일 환경에서 심장병 환자를 위한 편재형 헬스케어 시스템의 구현," 한국컴퓨터정보학회논문지, 제 13권, 제 5호, 117-124쪽, 2008년 9월.
- [3] 이상민, 진평수, 조승호, 김형태, "노인의 편안한 생활을 지원하는 센서 통합 관리 시스템," 한국컴퓨터정보학회논문지, 제 15권, 제 1호, 103-109쪽, 2007년 6월.
- [4] 유수영, 김보원, 한승빈, 최진욱, 정재현, 이민경, 전종훈, "의료기관 정보공유를 위한 HL7 메시지 서버 프로토타입 설계 및 평가," 대한의료정보학회지, 제 7권, 제 3호, 2000년.
- [5] Orion System Symponia,
<http://www.orionhealth.com>
- [6] NeoTool Development, LLC,
<http://www.neotool.com>
- [7] Chameleon Messaging Toolkit,
<http://www.interfaceware.com/chameleon.html>
- [8] Eversolve, LLC,
<http://www.eversolve.com/products>
- [9] 엄기성, 조훈, 김일곤, 곽연식, "웹기반의 HL7 메시지 브라우저 개발," 대한의료정보학회지, 제8권, 제3호, 2002.
- [10] R.V. Sideli, S. Sengupta, I. Chernizer, G. Hripcsak, "An HL7 Toolkit," IEEE Computer Society Press, Vol. 17, pp. 904, 1993.
- [11] Z. Qing, G. Shangkai, "Design and Realization of HL7 Gateway," Biomedical Engineering, Vol. 20, No. 1, pp. 111-115, 2003.
- [12] Health Level Seven, <http://www.hl7.org>
- [13] K. Ohe, S. Kaihara, "Implementation of HL7 to Client-Server Hospital information System (HIS) in the University of Tokyo Hospital," Journal of Medical Systems, Vol. 20, No. 4, pp. 197-206, 1996.
- [14] Robert Eckstein, "Java Swing," O'reilly Media, 2002.
- [15] Java Parser Generator,
<https://javacc.dev.java.net>
- [16] 송준아, 임영미, 홍귀령, "요양시설 치매노인의 배회행동," 대한간호학회지, 제38권, 제1호, 2008년 2월.
- [17] 김한곤, "대구시 노인들의 치매 실태 및 치매노인들의 생활 습관적 특성," 한국인구학, 제28권, 제2호, 2005.

저자 소개

항 득 영



1988년: 광운대학교 전자계산학과(학사)
 1990년: 광운대학교 전자계산학과(석사)
 1999년: 광운대학교 전자계산학과(박사)
 1990년~1994년: 전주기전대학 전자계산학과 조교수
 1994년~현재: 강원대학교 삼척캠퍼스 컴퓨터공학과 교수
 관심분야: 프로그래밍 언어, 컴파일러, 의료정보