

자바 언어 기반 디지털 방송용 MPEG-2 패킷 분석 시스템

김인희⁰ 황 준

서울여자대학교 컴퓨터학과
swuinhee@mail.swu.ac.kr⁰, hjun@swu.ac.kr

The Java-based MPEG-2 Packet Analyzing System for the Digital Broadcast

In-Hee Kim⁰ Jun Hwang

Dept. of Computer Science, Seoul Women's University

요 약

디지털 방송을 위한 Java 언어 기반 MPEG-2 TS(전송 스트림) 패킷 분석 시스템은 TS 패킷의 내용을 분석하고, PSI Table과 SI Table 데이터를 수집하여 분석 할 수 있도록 해준다. 구현된 패킷 분석 시스템은 PAT, PMT, NIT, CAT 등의 PSI Table과 BAT, NIT, EIT, SDT 등의 SI Table의 정보를 계층적으로 분류 시켜준다. 또한 그 안의 포함된 다양한 의미의 콘텐츠를 포함하는 여러 종류의 Descriptor들의 정보도 분석해준다. 패킷 분석 시스템은 멀티-쓰레딩과 편리한 데이터 관리를 위해서 몇몇 클래스들로 구성 되어 있다. 그리고 패킷 분석 시스템의 GUI는 Swing 라이브러리로 구현되었기 때문에, 시스템은 GUI의 변화 없이 윈도우즈, 리눅스와 같은 여러 가지 플랫폼에서도 안정적으로 동작된다. 본 연구의 시스템은 향후 DSM-CC 메카니즘과 의미적 분석의 구현 그리고 데이터베이스 시스템과의 연동 등으로 심화되어 연구될 것이다.

1. 서 론

디지털 방송은 기존 텔레비전 방송에서의 아날로그 비디오와 오디오 정보를 MPEG-2 형식으로 디지털화하여 다양한 전송 경로를 통해 시청자에게 제공하는 방식이다. 방송이 디지털화되면서 비디오, 오디오만으로 이루어진 기존의 텔레비전 방송에서 벗어나 디지털 데이터를 방송하는 것과 시청자와 콘텐츠 제공자가 양방향으로 정보를 교환할 수 있는 대화형 방송 등이 가능해졌다 [1,5,8]. 디지털 방송은 그 변조 방식에 따라 위성, 지상파 그리고 케이블 방송 등으로 나뉜다. 이 중 본 연구에서는 가장 널리 사용되고 있는 디지털 위성 방송의 전송 스트림을 분석하는 시스템을 개발하였다. 본 연구에서 개발한 디지털 방송을 위한 TS 패킷 분석 시스템은 MPEG-2 형식의 TS 스트림을 입력받아 패킷의 내용을 분석하도록 개발되었다 [2,3,4]. 구현된 TS 패킷 분석 시스템을 이용하면 MPEG 형식으로 이루어진 디지털 방송 콘텐츠를 자동으로 분석할 수 있게 되며, 디지털 방송을 실시간으로 모니터링 하여 그 특성을 자동으로 통계화할 수 있는 등의 이점이 있다. 또한 본 연구에서는 플랫폼 독립적인 Java를 이용하여 패킷 분석 시스템을 구현함으로써 시스템 동작의 안정성을 높이고자 하였다 [6].

2. 관련 연구

MPEG-2 전송 스트림 패킷 분석 시스템 혹은 DTV 모니터링 시스템은 현재 KBS, MBC와 같은 국내 주요 방송사업자들의 기술연구소에서 연구되고 있다 [7]. DVB 위성 방송용 TS 분석 시스템뿐만 아니라, ATSC 지상파

방송용 스트림 모니터링 시스템도 포괄적으로 연구 개발되고 있다. 이러한 DTV 모니터링 시스템은 디지털 방송의 송출, 수신 및 스트림의 이상 유무를 실시간으로 모니터링 할 수 있어 디지털 방송의 안정성에 기여할 수 있다. 이들 방송사업자들은 현재 TS 모니터링 시스템과 위성 중계기의 RF 신호 모니터링 시스템의 통합 그리고 네트워크 구축을 통한 다 수의 지역 DTV 모니터링 시스템들과 데이터베이스 시스템과의 통합을 적극적으로 추진하고 있다.

또한 최근에는 디지털 멀티미디어 데이터 방송(대화형 방송)에 관심이 많아져서, MPEG-2 전송 스트림 기반의 디지털 데이터 방송을 위한 DSM-CC(Digital Storage Media - Command and Control) 스트림도 모니터링하여 제어하는 연구도 활발하게 이루어지고 있다.

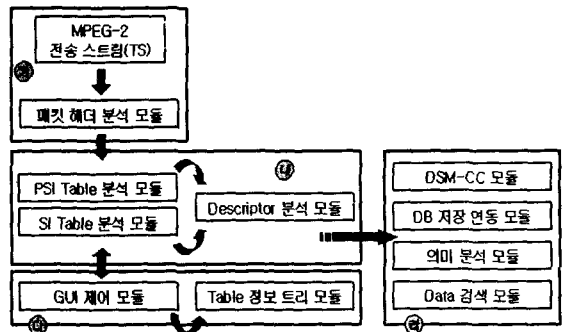


그림 1 TS 패킷 분석 시스템

3. 시스템의 구조

본 논문에서 개발한 디지털 방송을 위한 TS 패킷 분석 시스템은 MPEG-2형식의 TS Stream을 입력받아 각 TS 패킷을 PID별로 구분하여 PSI Table, SI Table과 Descriptor들의 정보를 수집 및 분석 할 수 있도록 제작되었다. 플랫폼 독립적인 Java 언어를 이용하였고, 사용자 편리성을 고려하여 GUI(Graphic User Interface)는 Java의 Swing 라이브러리를 이용하여 구현하였다.

3.1 시스템 구조 개요

본 논문의 TS 패킷 분석 시스템은 대략 그림 1과 같은 구조로 구성되어 있다. (가) TS 스트림 입력부: MPEG-2 형식으로 압축된 TS 파일을 통해 입력된 스트림 데이터는 패킷 헤더 분석 모듈을 통해 실시간 병렬처리로 PID별 패킷들로 분류되어 분석된다. (나) 패킷 데이터 분석부: (가)단계에서 분류된 패킷은 PSI Table과 SI Table 그리고 PES 등으로 분류되어 각 분석 모듈에 의해 의미 정보 필드별로 파싱(Parsing)된다. 여기서 Table내 Descriptor들은 별개의 분석 모듈에서 파싱하여 처리된다. (다) GUI관련 제어-표시부: (가), (나)단계에서 실시간 병렬처리로 얻어지는 패킷 분석 결과를 사용자 인터페이스에 출력하고, 특히 Table 정보를 패킷과 연관이 되어 트리구조로 계층화하여 출력한다. (라)의 DSM-CC 모듈 등은 현재 구현된 시스템에 향후 연결되어 동작할 구현 예정인 기능 모듈들을 나타내었다.

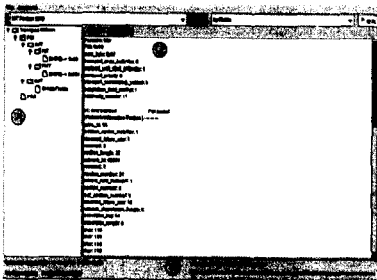


그림 2 TS Packet Analyzer의 GUI

3.2 시스템 GUI 구조

Java Swing 라이브러리로 설계된 GUI는 그림 2와 같다. (가) PSI, SI Table 분류 표시부에서는 현재 시스템에 입력되는 TS 스트림의 패킷의 PID를 참조하여 PSI Table, SI Table 그리고 프로그램을 담고 있는 PES 등으로 분리하여 계층 구조로 표시해준다. 사용자는 트리의 각 항목을 간단히 선택함으로써, 각 Table과 함께 표시되는 Packet 관련 정보를 통하여 해당 패킷 정보를 바로 알아 볼 수 있게 된다.

(나) 패킷 내용 표시부에서는 시스템에 현재 입력되는 Packet의 헤더정보와 PSI, SI Table 및 Descriptor 등의 상세한 필드 정보를 나타내주는 영역이다. 사용자가

Packet의 내용을 필드 단위부터 바이트 단위, 비트 단위 까지 선택하여 상세하게 모니터링 할 수 있도록 제작되었다.

(다) TS 스트림 제어부에서는 시스템에 입력되는 TS Packet 스트림의 진행 정보를 표시하고, 특정 Packet을 저장하거나, 원하는 시점에서의 Packet 정보를 사용자가 접근할 수 있도록 설계되었다.

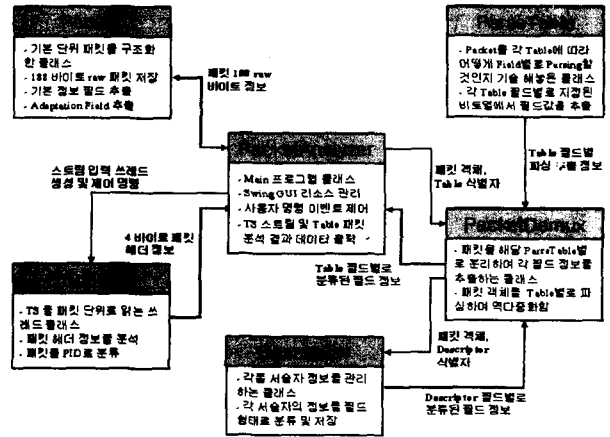


그림 3 Java로 구현된 패킷 분석 시스템의 클래스 구조

3.3 시스템 클래스 구조

TS 패킷 분석 시스템은 실시간 분석의 효율성 향상과 플랫폼 독립성을 확보하기 위해서 객체 지향 언어인 Java로 구현되었다. TS 패킷 분석 시스템은 대략 그림 3과 같은 클래스 구조로 구성되어 있다.

'TS_Read'클래스는 TS 스트림을 패킷 단위로 읽는 스레드(Thread) 클래스이다. 입력되는 TS 패킷을 처리하기 위해 스레드 클래스를 상속받아, 해당 TS 스트림을 188 bytes의 패킷 단위로 빠르게 건너뛰며 헤더 정보를 분석한다. 널(Null)패킷을 제외한 패킷을 PID를 기준으로 분류하여 Packet 객체로 저장하고, 특정 Table 정보를 담고 있는 패킷 발견 시 PacketDemux 객체를 통하여 해당 Table 정보를 파싱(Parsing) 하도록 해준다.

'Packet'클래스는 TS 스트림의 기본 단위인 패킷을 구조화한 클래스로 188 bytes의 패킷 정보 그대로를 저장하게 된다. 4 bytes의 헤더 정보 이외에, 패킷의 기본 필드 정보(대략 sync byte에서 continuity counter 까지)와 적응 필드(Adaptation field) 정보를 실시간으로 모니터링 할 수 있게 해준다.

'ParseTable'은 주어진 Packet 객체를 각 Table에 따라 어떻게 Field별로 Parsing할 것인지 기술 해놓은 클래스이다. 해당 Table의 각 필드별로 해당 비트수와 필드명을 배열로 저장하여 입출력하게 되어있다. 각 SI Table의 필드별로 지정된 비트열에서 해당 값을 추출하여 주어진 패킷을 파싱하여 PacketDemux에 파싱 정보를 넘겨준다.

'PacketDemux'는 주어진 임의의 패킷을 해당 ParseTable별로 분리하여 각 필드 정보를 추출하는 패킷

디코더(Decoder) 클래스이다. Table별로 일괄 번호를 매겨 ParseTable을 생성하여 관리하며, 패킷 객체를 Table별로 파싱하여 해당 정보를 GUI에 출력하도록 해준다.

'Descriptor'는 PSI와 SI의 각종 정보를 나타내는 서술자의 내용을 필드 단위로 파싱하여 입출력하는 클래스이다. SI Table마다 포함하고 있는 서술자들의 종류는 조금씩 다르며, 최대 250여개 정도의 서술자가 사용된다. ParseTable과 마찬가지로, PacketDemux에 필드별로 파싱된 정보를 넘겨준다.

'PacketAnalyzer'는 프로그램의 메인 클래스로 Swing 라이브러리의 JFrame을 상속받았다. 프로그램의 GUI 리소스들을 관리하며, 사용자 명령과 각종 이벤트들을 관리한다. TS 스트림 패킷 및 PSI, SI Table 분석 결과 등의 데이터 출력을 제어한다.

4. 시스템 환경 및 동작 분석

본 연구에서 개발한 디지털 방송을 위한 TS 패킷 분석 시스템은 MS Windows XP, JDK 1.4.1, Swing GUI 환경에서 개발되었다. 여러 개의 독립적인 Java 클래스 구조로 구성된 본 연구의 패킷 분석 시스템은 대략 다음과 같은 절차로 동작한다.

| | |
|---|--|
| <pre>Packet = 60* PID: 0x10 - NIT sync_byte: 0x47 transport_error_indicator: 0 payload_unit_start_indicator: 1 transport_priority: 0 transport_scrambling_control: 0 adaptation_field_control: 1 continuity_counter: 11 ... 47 41 11b 02 b0 52 01 c7 0 0 ff ff 70 0 b e 1 1 f 0 1 f 13 5 0 0 0 a 0 14 d 0 22 0 0 8 ff ff ff ff ff ff ff ff 52 1 22 66 4 0 f 0 1 c e 6 1 f 0 12 14 d 0 64 0 0 8 ff ff ff ff ff ff ff ff 52 1 64 5 e 5 1 f 0 5 6 f 3 0 1 ff 6b 2a 29 b ff ff ff ff ff ff ... 01000111 01000000 00000000 00011100 00000000</pre> | <pre>[Program Association Section] ----- table_id: 0 section_syntax_indicator: 1 '0': 0 reserved: 3 section_length: 17 transport_stream_id: 257 reserved: 5 version_number: 24 current_next_indicator: 1 section_number: 1 last_section_number: 0 program_number: 0 reserved: 7 network_pid: 16 (0x10) program_number: 1 reserved: 7 program_map_pid: 257 (0x101) PAT loop N=2 ... [Network Information Section] ----- table_id: 64 section_syntax_indicator: 1 reserved_future_use: 1 reserved: 3 section_length: 38 network_id: 65281 reserved: 5 version_number: 24 current_next_indicator: 1 section_number: 0 last_section_number: 0 reserved_future_use: 15 network_descriptor_length: 6 descriptor_tag: 64 descriptor_length: 4 char: 116 'f' char: 101 'e' char: 115 's' char: 116 's' reserved_future_use: 15 transport_stream_loop_length: 19 transport_stream_id: 257 original_network_id: 65281</pre> |
|---|--|

그림 4 필드별, 바이트별, 비트별로 표시된 패킷 정보의 예

PacketAnalyzer를 통한 사용자의 임의의 SI Table에 대한 모니터링 요청으로 인해, TS_Read는 쓰레드로 생성되며, 실시간으로 TS 스트림을 입력받아 패킷의 헤더 정보를 파악한다. 패킷의 헤더 정보 중 PID 필드 정보를

통하여 각 패킷은 우선적으로 PSI, SI Table용 패킷 혹은 PES 패킷으로 분류된다. SI Table에는 DVB 스펙에 명시된 PAT, PMT, NIT, CAT 등이 있다[2,3].

만약 사용자가 NIT SI Table을 모니터링 하라고 시스템에 요구했다면, TS_Read에서는 입력 스트림에서 패킷들의 PID를 검색하여 NIT에 해당하는 값을 가진 패킷을 'Packet' 객체로 저장 후, PacketDemux로 넘겨준다.

PacketDemux는 넘겨진 Packet 객체를 NIT ParseTable 객체를 통하여 의미 필드별로 파싱한다. ParseTable은 파싱 과정에서 Descriptor를 만날 때 마다 그 종류를 파악하여 Descriptor 객체에 그 정보를 전달하여 Descriptor내의 필드 정보를 파싱할 수 있도록 해준다.

이러한 과정으로 실시간으로 파싱된 패킷안의 정보들은 필드별로 의미를 가진 SI Table 정보로 GUI에 체계적으로 표시된다. 그림 4는 패킷 정보가 필드별, 바이트별, 비트별로 표시된 예를 나타내며, 그림 5는 PAT, NIT Table을 모니터링 하는 경우 본 연구의 패킷 분석 시스템을 통하여 도출된 결과 값의 예를 나타낸 것이다.

5. 결론 및 향후 과제

본 연구에서 구현한 디지털 방송을 위한 Java 언어 기반 MPEG TS 패킷 분석 시스템은 실시간으로 각종 Descriptor들을 포함한 PSI, SI Table 데이터를 수집하고, 분석할 수 있도록 해준다. 제안된 패킷 분석 시스템은 멀티-쓰레딩과 편리한 시스템 데이터 관리를 위해서 몇몇 독립적인 클래스들로 구성 되어 있다. 패킷 분석 시스템의 GUI는 Swing 라이브러리로 구현되어, 시스템은 GUI의 변화 없이 윈도우즈, 리눅스와 같은 여러 플랫폼에서도 안정적으로 동작된다.

향후 본 연구의 시스템은 리눅스 플랫폼으로의 시스템 포팅, 디지털 데이터 방송용 DSM-CC 및 의미 해석기 (Semantic Interpreter)의 구현, 그리고 데이터베이스 시스템과의 연동 등으로 확장되어 연구될 예정이다.

참고 문헌

- [1] 김대호, "양방향 TV", 나남출판, 2002.
- [2] Specification for Service Information in DVB systems, ETSI EN 300 468 V1.4.1, 2000.
- [3] 유시룡 외, "MPEG 시스템", 브레인코리아, 2000.
- [4] ISO/IEC 13818-1 - Information Technology - Generic Coding of Moving Pictures and Associated Audio : Systems, Recommendation H.222.0, 1996.
- [5] 오덕길, 김호겸, "대화형 위성통신 방송 기술", 한국 전자통신 연구원, 2001.
- [6] Sun Microsystems, Inc., "Java 2 Platform, Standard Edition, v 1.4.1 API Specification", 2002.
- [7] 이상주, "DTV 모니터링 시스템", 제7권, 방송공학회지, 2002.
- [8] 안병호, "AIC 기반 대화형 TV 구현에 관한 연구", 광운대 석사학위논문, 2000.