

# 다운로드 가능한 전자프로그램가이드의 구현

오단비  
아카넷TV

정문열  
서강대학교 영상대학원 미디어공학과  
[juliana@acanettv.com](mailto:juliana@acanettv.com) [moon@sogang.ac.kr](mailto:moon@sogang.ac.kr)

## Implementation of Downloadable EPG

DanBi Oh  
Acanettv Corp

MoonRyul Jung  
Dept of Media Technology  
Graduate School of Media Communications  
Sogang University

### 요약

디지털방송은 다수의 채널 공급을 가능하게 하는데, 이러한 다채널 환경에서는 각 채널로 일일이 전환하지 않고도 어느 채널에 어떤 정보가 방송되고 있는지를 일목요연하게 파악 할 수 있는 전자 프로그램 가이드(EPG: Electronic Program Guide)가 필연적으로 요구된다. 본 논문은 EPG의 한 구현방법으로서 방송을 통해 TV에 전달되어 실행되는 Downloadable EPG를 설명하고 우리나라 디지털 위성방송 표준인 DVB-MHP 규약을 따르는 구현방법과 구현결과를 기술하였다. 본 논문은 데이터 방송과 EPG의 구현에 관련된 참고 기술자료가 미비한 현실에서 디지털 방송의 EPG 개발자들이나 응용프로그램(application) 개발자들이 DVB-MHP 규약에 따라 다양한 종류의 EPG를 개발하는데 도움이 될 것이다.

### 1. 서론

디지털 방송은 기존 아날로그 신호의 디지털 화를 통해 화질음질을 향상시키고 방송에 양방향성을 부여할 수 있는 데이터방송 (Interactive Service, iTV) 을 가능하게 하였다. 그리고 압축에 의해 아날로그 방송보다 적은 대역 (Bandwidth)을 사용하다 보니 다채널 방송이 가능하게 되었다. 현재 외국의 디지털 위성방송의 경우, 서비스 채널의 수가 수백에 달하고 있고 그 수는 더욱 늘어날 전망이다. 이런 다채널 환경에서는 시청자가 각 채널을 모두 둘러보지 않고도 어느 방송 채널에서 어떤 정보가 제공되고 있는지를 쉽게 파악할 수 있는 기능이 요구되는데, 이를 위해 제공되는 서비스가 EPG (Electronic Program Guide)이다. EPG는 시청자가 가장 빈번하게 사용할 것으로 기대되는 TV의 서비스라 할 수 있으며, 채널 수가 많아질수록 그 필요성은 더욱 커진다. 데이터 방송 서비스 중에 확실한 수용자 욕구가 예상되는 것이 EPG 이다. 또한TV의 대중성과 범용성을 고려할 때EPG를 통해 창출되는 잠재적인 부가가치는 PC의 포털 서비스 이상이 될 수 있을 것이다.

EPG는: STB(Set-Top Box, 디스플레이에 연결되어 디지털 방송을 수신할 수 있게 하는 장치로 디지털 TV인 경우 안에 내장되어 있는 내장형 EPG와 방송에 포함되어서 플랫폼(Platform)사업자로부터 송출되어 각 가정의

STB로 전송되는 다운로드 가능한 downloadable EPG로 분류될 수 있다. Downloadable EPG는 플랫폼사업자나 채널 사업자가 방송한 EPG 프로그램을 STB 가 수신하여 시청자에게 서비스해주는 방식으로 내장형 EPG에 비해 새롭고 다양한 서비스를 가능하게 한다.본 논문은 DVB-MHP 표준을 기반으로 하는 다운로드 가능한 (downloadable)EPG의 구현에 관한 논문으로 그 구현 방법을 구체적으로 기술 한다.

### 2. EPG

EPG는 방송스트림으로부터 사용자가 특정 채널을 선택하도록 도와주는 어플리케이션(application)이다. 따라서 EPG에 대한 논의를 하기 위해 디지털 방송에서 채널이 어떻게 선택되는지를 우선 개략적으로 설명한다.

#### 2.1 채널 선택 메카니즘

DVB에서는 아날로그방송에서 채널에 해당하는 개념을 서비스(Service)라 하고 하나의 서비스안에 여러 개의 성분 스트림 (다수의 비디오, 다수의 오디오, 다수의 데이터 스트림)을 포함시켜 방송할 수 있다. 물론 이들 다수의 스트림이 하나의 공통적인 서비스를 구성하는 요소들 일 경우에만 하나의 서비스로 묶을 필요가 있다. 그리고 우리가 흔히 말하는 방송 프로그램 (9시 뉴스, 한일전의 전반전)을 이벤트(Event)라고 한다. 하나의 이벤트는 시

작시각과 끝나는 시각이 있으며, 다수의 성분 스트림을 포함할 수 있다.

STB는 디폴트 주파수를 가지고 있어서 STB가 구동 되면 STB의 튜너가 디폴트 주파수에 에 튜닝을 하여 해당 되는 TS(Transport Stream)를 읽는다. 디지털 방송 스트림은 다수의 TS로 구성되어 있다. 한 TS에는 모든 TS에 대한 기본적인 정보와 이들 TS에 포함되어 있는 서비스 및 이벤트에 대한 정보들이 모두 포함되어 있다. 따라서 사용자가 원하는 채널이 디폴트 TS안에 없으면 이 TS안에 있는 정보를 이용하여 원하는 채널(서비스)이 포함되어 있는 TS에 튜닝 할 수 있다.

STB가 읽고 있는 TS안에 포함된 모든 채널(서비스)에 접근하기 위해서는 우선 이들에 대한 로케이터(Locator)들을 먼저 얻어야 한다. 서비스 로케이터는 서비스를 가리키는 URL 같은 것이다. 서비스 로케이터는 Original Network id, Transport Stream id, Service id 로 구성되어 있다. 특정 서비스 로케이터가 가리키는 서비스안에 있는 프로그램들(이벤트)을 가리키는 이벤트 아이디도 있다. 따라서 서비스 로케이터와 이벤트 아이디를 더하면 이벤트 로케이터도 얻을 수 있다. DVB-MHP 는 TS로부터 이 안에 들어있는 이벤트들의 로케이터를 확보할 수 있도록 하는 API를 정의하고 있다. 가장 원시적인 EPG는 이런 API를 이용하여 모든 이벤트 로케이터와 사용자가 이벤트를 식별하는 데 도움을 주는 이벤트 이름들을 확보한 후, 이벤트 이름을 사용자에게 보여주고 사용자로 하여금 원하는 이벤트 이름을 선택하게 하는 것이다. EPG는 선택된 이벤트 이름에 해당하는 이벤트 로케이터를 이용하여 TS안에 포함된 해당 이벤트를 가져와 이를 셋톱박스의 스트림 플레이어에게 전달한다.

## 2.2 EPG의 정의 및 분류

EPG(Electronic Programming Guide)란 시청자가 각 채널로 일일이 전환을 하여 내용을 확인하지 않고도 어느 채널에서 어떤 내용이 방송되고 있는지를 파악할 수 있는 방법을 전자적으로 구현한 서비스이다. 이를 위해 EPG는 방송물에 대한 정보를 획득하고, 분류하여 보여주며, 탐색이나검색을 통하여 시청자가 원하는 방송물을 선택하면 그 방송물을 화면에 보여주는 역할을 한다.

현재 한국디지털위성방송 (스카이라이프)에서 제공되는 EPG는 STB (settop box) 가 만들어질 때부터 그 안에 상주해있는 내장형 EPG(embedded EPG)이다. 내장형 EPG는 안정적이고 통일된 사용자 인터페이스를 제공한다. 내장형 EPG는 플랫폼 사업자 혹은 STB 제조사가 독자적인 방식으로 구현한 것으로서 그 구현방법이 일반에 공개되어 있지 않아 EPG에 다양한 기능을 부가하거나 EPG를 통해 새로운 사업모델(business model)을 시도할 기회가 제한된다. TV의 대중성과 범용성을 고려할 때 다양한 EPG와 부가적인 서비스의 개발과 보급은 필연적인데 이를 위해 공개적이고 서비스 사업자의 접근이 용이한 EPG 서비스의 개발이 필요하다.

그 대안으로서 고려될 수 있는 방법이 방송으로 수신되

어 다운로드 가능한 EPG(downloadable EPG)이다.

## 2.3 Downloadable EPG

Downloadable EPG는 플랫폼 사업자나 채널사업자가 EPG를 일반 방송 채널을 통해 전송하고, 사용자가 STB에서 일반 채널 선택하듯이 EPG 채널을 선택한다. Downloadable EPG를 제공하는 사업자는 방송규격에만 맞추어 EPG 프로그램과 데이터를 전송하면 되므로 다수의 사업자가 다양한 EPG를 시도하고 새로운 비즈니스 모델과 연계하여 서비스할 수 있는 기회가 제공된다. 본 논문에서는 한국 위성방송의 표준이고 유럽의 공개 방송 표준인 DVB-MHP를 기반으로 한 downloadable EPG를 기술한다.

Downloadable EPG 서비스를 위해서는 EPG 어플리케이션과 방송물 관련 데이터(EPG 데이터)를 방송해야 하는데, DVB-MHP 환경 하에서 어플리케이션은 xlet이라고 불리는 특수목적 Java 프로그램을 의미한다. 이는 데이터캐로셀(Data Carousel)로 인코딩된 후 디지털 방송의 전송스트림인MPEG-2 TS속에 포함되어 방송된다. DVB-MHP 환경에서 EPG 데이터를 방송하는 방법은 크게

세가지로 나눌 수 있다. 첫째는 DVB-SI (Service Information)의 서비스 및 이벤트 정보에 포함되어 있는 방송 프로그램에 대한 정보를 이용하는 것이다. 둘째는 DVB-SI는 전혀 사용하지 않고 서비스 및 이벤트 정보를 별도의 데이터 캐로셀로 전송하여 그 정보를 이용하는 것이다. 셋째는 위 두 방법을 혼합하여 DVB-SI 정보는 물론 데이터캐로셀로 전송되는 추가 정보도 함께 이용하는 방법이다. 본 논문에서는 첫번째 방법을 이용하였는데, 플랫폼 사업자가 각 채널 사업자로부터 서비스 및 이벤트 정보를 제공 받아 SI Generator를 통해 DVB-SI로 변환한 후 MPEG-2 TS 에 포함시켜 방송하는 EPG 데이터인 DVB-SI를 사용하는 EPG 애플리케이션을 개발한다.

## 3. 디지털 방송 시스템의 구조

Downloadable EPG는 디지털 방송에서 데이터 방송이 가능해 짐에 따라 가능해 진 것이므로 우선 데이터 방송의 구조에 대해 간단히 살펴본다. EPG프로그램을 구현하기 위해서는 디지털 방송에서 전송되는 서비스와 이벤트에 대한 정보를 어떻게 표현하고 어떻게 추출하는지를 알아야 한다. 따라서 이런 정보는 정보에 대한 정보로서 미리 정의된 포맷에 따라 표현되고 전송된다.

### 3.1 PSI (Program Specific Information)

TS에는 다수의 서비스와 이 안의 다수의 이벤트에 대한 정보가 포함되어 전송된다. 이런 정보를 위해 디지털 방송 전송스트림 표준인 MPEG-2 System에서는 PSI(Program Specific Information)를 정의하고 있다. PSI는 방송 프로그램이 어떻게 구성되어 있는지를 나타

내는 정보로서 TS로부터 방송 프로그램을 읽어내기 위해 쓰인다.

PSI정보는 PAT(Program Association Table), PMT(Program Map Table), CAT (Conditional Access Table), NIT(Network Information Table)의 4개의 테이블로 구성된다. PAT는 TS에 포함된 각각의 서비스에 대한 정보인 PMT를 가리키고 있고, PMT는 특정 서비스에 포함된 성분 스트림 (component streams, 개개의 비디오, 오디오, 또는 데이터 스트림을 지칭 함) 에 대한 정보를 가지고 있다. NIT는 전송 네트워크에 대한 정보, 그리고 CAT는조건부 수신(Conditional access)이 필요할 경우 스크램블링 등을 위한 정보를 포함한다.

STB가 특정 TS에 튜닝 되면, STB내의 Service Tuner가 우선적으로 PAT를 읽는다. 그리고 여기서부터 시작하여 특정 채널(플랫폼 사업자가 정해 놓은 서비스)를 선택 방송을 한다. 일반적으로 디폴트 채널은 플랫폼 사업자를 선전,소개하는 방송을 하는 경우가 많은데, EPG 채널을 방송할 수도 있다.

### 3.2 DVB SI (DVB Service Information)

서비스 정보는 수신기에서 프로그램을 수신하여 Tuning 및 프로그램 가이드를 위한 정보를 포함하는 디지털 방송에서의 기본 서비스 정보 규격이다. 서비스 정보의 내용으로는 Audio/Video 편성정보 및 채널 정보, CAS(Conditional Access System) 정보 등 EPG를 위해 필요한 여러 가지 정보를 포함하고 있다.

DVB에서는 아날로그방송에서 채널에 해당하는 개념을 서비스(Service)라 하고 이 안에서 여러 개의 방송 프로그램을 전송할 수 있다. 방송 프로그램을 이벤트(Event)라고 한다.

DVB-SI는 방송되는 스트림에 대한 정보를 기술하고 있는 데이터로 MPEG-2 시스템 규약에서 정의한 스트림에 대한 정보인 PSI 이외에 DVB에서 추가로 정의한 정보이다. 이 정보는 테이블들로 구성되어 'SI TABLE' 이라 불리는데, SDT, EIT, NIT 등이 있다.

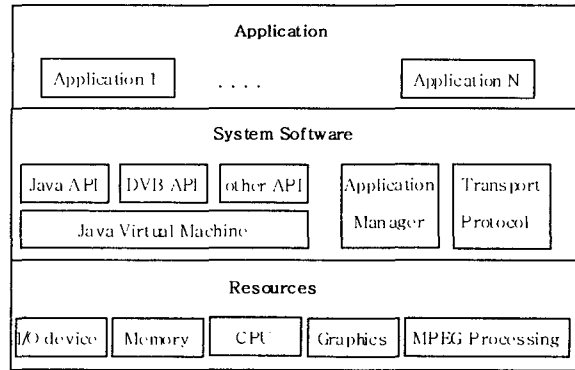
SDT는 방송되는 서비스들에 대한 정보를 제공하는데, 서비스 제공업자와 서비스의 이름 등에 관한 내용을 전달한다. 사용자가 하나의 서비스를 선택할 때 이것이 현재 STB가 튜닝하고 있는 TS안에 들어 있을 수도 있고, 다른 TS에 들어 있을 수도 있다. 후자의 경우, STB는 이 새로운 TS의 주파수를 이용하여 새로이 튜닝해야 한다.

EIT는 서비스 내의 개별 이벤트들의 이름과 시작 시간, 방영기간 등과 같은 정보로 구성되는데, 현재 방영되거나 다음에 방영될 이벤트에 대한 정보와 이벤트 스케줄 정보를 전달한다. 이 정보는 현재 TS의 서비스에 국한된 정보일 수도 있으며 다른 TS의 이벤트에 대한 정보일 수도 있다. NIT는 MPEG-2 PSI에 정의되었지만, MPEG-2 표준에서는 NIT를 구성하는 내용을 구체적으로 정의하지 않았기 때문에, DVB에서 NIT의 세부 내용을 재 정의하였다. TS를 유일하게 구분 짓기 위해,

transport\_stream\_id와 original\_network\_id 의 결합을 사용한다. 네트워크에는 개별적인 코드번호 network\_id가 주어지며 각 network\_id값은 ETR 162에 규정된다.

### 3.3 셋톱박스 소프트웨어의 계층구조

디지털 방송의 특징은 다채널과 고화질 서비스를 제공할 수 있으며, 또한 데이터 방송(Data Broadcasting)이 가능하다는 점이다. 데이터 방송은 데이터를 오디오, 비디오와 함께 방송하여 시청자에게 전달하는 것을 의미하는데, 이를 위한 국제 표준으로는 북미의 ATSC-DASE와 OCAP, 유럽의 DVB-MHP등이 있다. 본 논문의 EPG는 우리나라의 디지털 위성방송 표준인 DVB-MHP를 기반으로 구현되었다. 그림 1은 DVB-MHP 규약을 따르는 셋톱박스의 구조를 나타내고 있다.

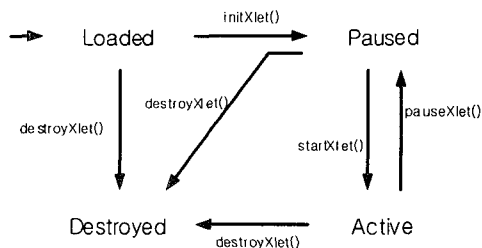


[ 그림 1. 셋톱박스의 소프트웨어 계층구조 ]

DVB-MHP는 어플리케이션을 전송스트림에 포함시키고 STB에서 다시 읽어내는데 필요한 프로토콜을 정의하고 있다. 어플리케이션은 어플리케이션 매니저의 통제를 받으면서 Java 가상기계에 의해 수행된다. 어플리케이션은 시스템 자원을 직접적으로 접근하지 못하며, 대신 시스템 소프트웨어(미들웨어)를 통해서 한다.

### 4. Xlet을 이용한 EPG 구현

본 논문의 EPG는 Java로 구현된 어플리케이션(Application)이다. TV용 Java 어플리케이션을 'Xlet' 이라고 하는데, Xlet은 [그림1]의 System software내의 Application Manager를 통해 그 lifecycle 이 제어되는데, Xlet의 lifecycle은 웹브라우저에서 실행되는 자바 어플리케이션인 applet(애플릿)과 같이 loaded, paused, active, destroyed의 상태(state)를 가지고, applet의 inti(), start(), pause(), destroy()에 해당하는 intiXlet(), startXlet(), pauseXlet(), destroyXlet() 메소드(method)를 가진다.



[ 그림 2. Xlet의 생명주기 ]

#### 4.1 EPG Xlet의 시작

EPG Xlet을 실행하기 위해 Application Manager는 먼저 EPG Xlet 객체의 initXlet()을 호출해야 한다.

initXlet() 메소드는 EPG 객체의 실행에 필요한 초기화 작업을 수행한다. 먼저 EPG를 구현하는 모든 클래스들의 객체를 생성하고, 각 객체에 필요한 초기화 작업을 한다. 초기화 작업 중의 하나로 Xlet 프로그램이 처리할 이벤트들을 선택하고 각 이벤트가 발생할 때 이를 처리할 이벤트 리스너 클래스의 객체를 생성하고 이 객체가 해당 이벤트의 리스너 객체임을 등록한다. 그리고 이벤트 리스너 클래스는 해당 이벤트를 처리하는 메소드(핸들러)를 정의해야 한다. 이벤트가 발생하면 이 이벤트와 연결된 리스너 클래스의 객체로 이벤트가 전달되고 이 객체는 이벤트를 처리할 핸들러를 이벤트를 입력인자(input parameter)로 하여 호출하게 되어 있다.

또 초기화 작업에서 해야 될 일은, EPG에 필요한 채널 정보, 프로그램 정보 등에 관련한 SI(Service Information)를 추출(retrieval)하기 위한 준비작업과 디스플레이 될 EPG 이미지(image)들을 로딩>Loading)시켜 놓는 것이다. 준비작업은 셋톱박스로 들어오는 모든 서비스들에 대한 TS 로케이터 리스트를 얻어내고<sup>1)</sup>, 서비스 및 이벤트 정보를 추출해 내기 위해서 SI table들을 관리해주는 최상위 클래스인 SIDatabase class<sup>2)</sup>의 객체를 획득하는 과정을 뜻한다. 이러한 초기화 작업이 끝난 다음 Application Manager는 startXlet()을 호출한다. startXlet()에서는 로딩시켜 놓은 이미지와 \*\*\*추출해 놓은 SI를\*\* (어떤 SI를 추출했는데? startXlet()에서는 SI 추출할 준비한 한 것이 아니었는가?) 가지고 EPG를 화면에 디스플레이한다.

#### 4.2 EPG Xlet의 진행

Xlet 프로그램이 Java로 구현되어 있고, 프로그램의 실행이 Java의 이벤트 구동(event driven)방식을 이용하기 때문에 EPG의 진행은 해당 이벤트를 처리하는 이벤트 핸들러에 의해 이루어진다. 이벤트 처리는 Xlet이 active 상태에 있을 때 이루어진다. EPG Xlet에서 진행되면서 실행되는 이벤트 처리기들은 다음과 같다.

- ◆ KeyListener - 사용자가 리모트 컨트롤 입력을 수행하면 실행.
- ◆ SIRetrievalListener - DVB-SI 데이터 추출이 완료되면 실행.
- ◆ SIMonitoringListener - DVB-SI데이터가 변경되면 실행

##### 4.2.1 Key Event 리스너

사용자가 리모트 컨트롤 키를 누르면 KeyListener의 KeyPressed()함수가 호출된다. KeyPressed()함수의 인자는 눌러진 키에 대한 정보를 담고 있는데, 눌러진 키

1) org.davic.net.tuning Package에서 제공하는 StreamTable Class의 listTransportStreams 메소드를 이용하여 얻어낸다.

2) MHP API의 org.dvb.si Package에서 제공하는 Class로, SI Table들을 관리 하기 위한 최상위의 class이다.

의 의미에 해당되는 처리를 미리 정의된 이벤트 처리기를 실행함으로써 한다.

##### 4.2.2 SI Retrieval Event 리스너

본 절에서는 채널 정보를 포함하고 있는 SIService와 프로그램 정보를 포함하고 있는 SIEvent, 그리고 이 프로그램에 대해 부가정보를 포함하고 있는 디스크립터(Descriptor)를 추출(retrieval)해내는 이벤트의 처리 과정을 설명한다.

###### 4.2.2.1 SIService retrieval 과정

SIServiceManager Class는 SI정보를 요청하는 메소드와 SI 정보도착 이벤트가 발생했을 때, 이를 처리하는 이벤트 핸들러를 가지고 있다. SIService정보를 얻어내는 과정을 아래의 3단계로 간략히 서술 하였다.

###### 1. SIService 정보의 retrieval 요청

초기화 과정에서 확보한 TSLocator List에 속해있는 각 TSLocator안에 포함된 모든 채널(Service)과 관련된 정보를 요청한다.<sup>3)</sup>

###### 2. SIService 정보 이벤트 발생

요청된 SI정보가 준비되면 SIRetrievalEvent<sup>4)</sup>가 발생된다. Xlet이 TS로부터 SIService, SIEvent등 SI관련 정보 추출을 요청한 경우 정보를 추출할 수 있는 준비가 완료되는 시점에서 SIRetrievalEvent가 발생한다.

###### 3. SIService 정보 도착 이벤트 처리

SI정보도착 이벤트가 SISuccessfulRetrieveEvent<sup>5)</sup>일 경우에, getResult method를 이용하여 준비된 정보를 추출한다.

위의 과정을 구현하는 코드의 중요한 부분은 다음과 같다.

```
public class SIServiceManager implements
SIRetrievalListener {
/*1. SIService 정보 추출(retrieval)을 요청 */
private void requestSIServiceRetrieval(){
...
for(int i=0 ; i < dvbTsLocator.length; i++) {
try{ ...
siDatabase.retrieveSIServices(SIInformation.FROM_CACHE_OR_STREAM, dvbTsLocator[i], this,
dvbTsLocator[i].getOriginalNetworkId(),
dvbTsLocator[i].getTransportStreamId(), -1, null );
```

3) org.dvb.si Package의 SIDatabase Class의 retrieveSIServices() 메소드를 사용하여 SIService 정보 추출을 요청을 한다.

4) SIRetrievalEvent Class는 SILackOfResourcesEvent, SINotInCacheEvent, SIObjctNotInTableEvent, SIRequestCancelledEvent, SISuccessfulRetrieveEvent, SITableNotFoundEvent, SITableUpdatedEvent 이렇게 6개의 subclass들을 가진다. 이 6개의 이벤트 중에서 하나의 이벤트가 SI 추출 요청에 대한 응답 이벤트가 된다.

5) retrieve요청이 성공적으로 완료 되었을 때, 이 이벤트는 SI retrieval request으로 보내진다.

```

//"-1" 은 현재 TLocator안에 들어있는 모든 SI 정보
에 대한 요청한다는 뜻임.
}catch(SIIllegalArgumentException e) {
    e.printStackTrace();
    ...
}
/* 2.3.SIService 정보 도착 이벤트 발생에 따른 이벤트
처리*/
public void postRetrievalEvent (SIRetrievalEvent
event) { ...
    if(event instanceof SISuccessfulRetrieveEvent)
    {SIIterator iterator
    =((SISuccessfulRetrieveEvent)event).getResult();
    // 1) SIIterator에 추출된 SIService가 저장됨.
    // 2) 1)로부터 얻은 결과 값을 SIService Class
    형으로 변환.
    // 3) 2)를 통하여 얻은 SIService로부터 채널이름등
    채널에 관련된 정보 중 원하는 것을 얻어낸다.
    }
}

```

4.2.2.2. SIEvent retrieval 과정

SIEvent retrieval 과정은 4.2.2.1과정과 유사하므로 자세한 설명은 생략하도록 하겠다.

1. SIEvent 정보의 retrieval 요청

4.2.2.1 과정에서 확보한 Service Locator List에 속해있는 프로그램(SIEvent)과 관련된 정보를 모두 요청한다.<sup>6)</sup>

2. SIEvent 정보 이벤트 발생

3. SIEvent 정보 도착 이벤트 처리

위의 과정을 유사코드로 아래에 설명하였다.

```

public class SIEventManager implements
SIRetrievalListener{ ..
.
/*1. SIEvent 정보 추출(retrieval)을 요청 */
private void requestSIEventRetrieval() { ...
    for(int i=0; i< mSiServices.length; i++){
        mSiServices[i].retrieveScheduledSIEvent
        (SIInformation.FROM_CACHE_OR_STREAM,
        mSiServices[i].getDvbLocator(),
        new SIEventManager(this), null);
    }
//mServices[:4.2.2.1 과정에서 추출한 SIService들이
모두 저장되어있는 배열.
}
/* 2.3.SIService 정보 도착 이벤트 발생에 따른 이벤
트 처리*/
}

```

6) SIService Interface의 retrieveScheduledSIEvent 메소드를 사용하여 서비스 안에 scheduled events와 관련된 정보를 retrieve 한다. retrieval된 event들은 EIT-schedule안에 나타내어진 순서대로 표현되어 진다.

```

public void postRetrievalEvent (SIRetrievalEvent
event) { ...
    if(event instanceof SISuccessfulRetrieveEvent)
    {SIIterator iterator
    =((SISuccessfulRetrieveEvent)event).getResult();
    1)SIIterator에 추출된 SIEvent값이 저장됨.
    2) 1)로부터 얻은 결과 값을 SIEvent Class
    형으로 변환.
    3) 2)를 통하여 얻은 SIEvent 로부터 이름, 프로그램
    시작시간 등을 얻어낸다.
    } ...
}

```

4.2.2.1과 4.2.2.2 과정에서 SIService와 SIEvent정보를 추출하고 처리하는 EventHandler와 이를 디스플레이 하는 메소드사이에는 동기화(synchronization)가 필요하다. 또한 SIService정보를 추출하는 메소드와 SIEvent정보를 추출하는 메소드사이에도 동기화가 필요하다. 그 이유는 수시로 추출되는 SIService와 SIEvent정보들이 동시다발적으로 디스플레이됨을 방지하기 위해서, SIService retrieval이 완료 된 후에 SIEvent retrieval이 완료 되게 하고, 이러한 각각의 EventHandler의 작업이 모두 완료 되었을 때 화면에 채널과 프로그램 정보가 정확하고 균등하게 디스플레이 될 수 있도록 하기 위해서이다.

4.2.2.3 프로그램 부가정보 retrieval 과정

EPG 에서는, 채널(서비스)에 따라 각 프로그램(이벤트)에 대한 부가 정보(장르, 주연배우, 줄거리 요약 등)를 제공한다. 이런 부가 정보들은, EIT의 descriptor들을 통해서 전송할 수 있다. 장르에 대한 정보를 기술하는 descriptor는 content\_descriptor 이다. DVB-SI에서 지원하는 장르는 EN 300 468 [4]의 [Table18. Content\_nibble level 1 and 2 assignment]에서 정의하고 있다. Interface SIEvent의 getContentNibbles 메소드를 사용하여 이벤트의 장르를 획득할 수 있다. 두번째로, 이벤트 부가정보를 제공해주기 위한 descriptor는 Extended event descriptor 이다. Extended event descriptor는 이벤트의 이름 등이 기술되는 Short event descriptor에 추가하여, 이벤트에 관련된 세부적인 부가적인 정보들(ex. 드라마의 경우는 줄거리 요약이나 주연 배우 소개 등)이 기록되는 descriptor 이다. Extended event descriptor는 장르를 나타내어주는 content descriptor처럼 필수적으로 EIT에 꼭 있어야 하는 descriptor가 아니기 때문에 바로 이 정보를 획득할 수 있는 API는 정의 되어 있지 않으며, 개발자가 다른 API의 조합을 통해 구현을 해서 이 descriptor를 얻어내야 만 한다. 이를 구현하는 코드의 주요부분은 다음과 같다.

```

public class SIEventDescriptorManager implements
SIRetrievalListener
{ ...
/*1. SIEvent에 관련한 디스크립터 정보 추출

```

```

(retrieval)을 요청 */
public void retrieveDescriptorFromSIEvent
(SIEvent event) {
    short[] descriptorTags = new short[1];
    descriptorTags[0]
    = DescriptorTag.EXTENDED_EVENT;
//DescriptorTag클래스의 Field 값들 중에서, 원하는
descriptor의 필드 값 (여기서는 ExtendedEvent
Descriptor),을 얻어낸다.
    try{siRequest=event.retrieveDescriptors(
        SIInformation.FROM_CACHE_OR_STREAM,
        null, this, descriptorTags);
//4.2.2.2 과정에서 확보한 SIEvent에 속해있는
Descriptor정보를 모두 요청(descriptorTags는 윗 절에
서 얻어진 EXTENDED_EVENT의 Tag 이다.)
        ....
    }
/* 2.3. Descriptor 정보 도착 이벤트 발생에 따른 이
벤트 처리*/
public void postRetrievalEvent(SIRe trievalEvent
event){
    1) SIiterator에 추출된 결과들을 저장
    2) 1) 로부터 얻은 결과 값을
        Descriptor Class로 형으로 변환
    3)Descriptor 의 getContent s() 메소드를 사용하여
        내용을 추출해낸뒤 String으로 변환하여 저장.
}

```

#### 4.2.3. SIMonitoring Listener

EPG Xlet은 TS(Transport Stream)로 전송되어지는 SI가 바뀌는 경우를 모니터링 하고, SI가 갱신(Update) 되었을 경우 업데이트된 SI(채널정보, 프로그램 정보 등)를 화면에 디스플레이한다. SIMonitoring Event는 이와 같은 SI정보가 갱신되었음을 알리는 이벤트이다. 이 이벤트가 발생하면 SIMonitoringEvent Class에서 제공하는 getSource()를 통해 갱신(Update)된 SIDatabase를 얻을 수 있다.

#### 4.2.4. 채널전환

시청자가 많은 서비스들 중에서 특정 서비스 하나를 선택 하였을 때, EPG Xlet은 선택된 서비스에 대한 DvbLocator를 알고있다. EPG Xlet은 SIManager class의 getService() method로 하여금 선택된 DvbLocator를 이용하여 해당 Service를 얻어오고, ServiceContext Class의 select() method로 하여금 해당 서비스를 화면에 디스플레이하게 한다.

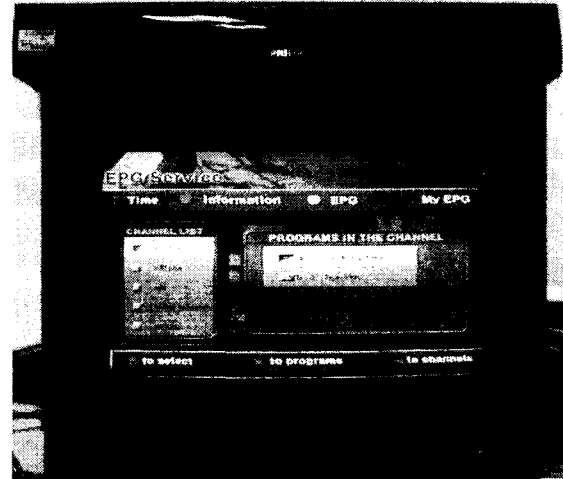
EPG Xlet은 새로운 서비스로 전환이 될 때, 새로운 서비스하에서도 계속 실행될 필요가 있다. 새로운 서비스의 AIT에 해당 xlet이 계속 실행되어야 함을 표시할 수 있으므로 방송을 제작하는 방송국에서 AIT를 적당히 생성함으로써 이 문제를 해결할 수 있다.

#### 4.3 EPG Application의 종료

Set-top Box가 꺼지면 EPG는Application manager가 내부적으로 destroyXlet()을 호출하여 EPG는 종료된다. EPG는 destroyXlet()이 호출되면 현재 진행 중인 작업을 모두 중단하고 EPG가 가지고 있는 모든 리소스(resource)를 반환(released) 해준다.

#### 5. 실험 및 결과

본 논문의 EPG는 개발자용 STB인 Philips TriMedia에서 개발되었다. 본 실험에서는 TS를 직접 STB 안의 하드 디스크에 저장하고 Flash Memory에 부팅 가능한(Bootable) 파일을 올려두어 실행시킴으로써 STB와 TV만으로 실제 위성으로부터, 수신하는 것과 같은 모의실험(Simulation)을 했다. 개발한 EPG를 PC로부터 STB로 Ethernet을 통해 download하면, EPG가 STB의 Middleware내의 Application Manager를 통해 실행된다.



[그림3. EPG가 DVB-MHP STB에서 실행되는 모습]

#### 6. 결론

본 논문에서는 디지털 방송 시대의 다 채널 환경에서 필수적인 EPG(Electronic Program Guide) 서비스 중에서 다양하고 새로운 서비스의 제공이 가능한 Downloadable EPG 서비스를 기술하고 우리나라 디지털 위성 방송 표준인 DVB-MHP 규격을 토대로 DVB-SI를 사용하여 구현한 방법을 설명하였다. 이를 위해 디지털 TV와 Downloadable EPG의 송출부터 수신까지의 개념 설명과 더불어, EPG구현에 필요한 MHPJava API 설명과 함께, 구현된 코드의 중요 부분을 설명하였다. EPG 구현에 관한 기술 자료가 미비한 현실에서 본 논문의 EPG 서비스 설계 및 개발 사례는 디지털 방송의 EPG 개발자들에게 실질적인 도움을 줄 수 있을 것이다.

#### 7. 참고문헌

- [1] DVB-MHP, <http://www.mhp.org>
- [2] ETSI, ETSI TS 102 812 Digital Video Broadcasting: Multimedia Home Platform(MHP) Specification1.1, Nov. 2001.
- [3] ETS 300 468 - Specification for Service

Information(SI) in DVB Systems.

[4] ISO/IEC 13818- 6: MPEG2 Digital Storage Media Command & Control -Part 6: Extensions for DSM-CC

[5] R. Chernock, R. Crinon, M. Dolan, and J. Mick. Data Broadcasting, McGraw-Hill, 2001.

[6] MPEG-2: Overview of the systems layer, Research and Development Report, BBC

[7] ISO/IEC 13818-1: "Information Technology Generic coding of moving pictures and associated audio-Part 1:Systems".

[8] ETR 162 Allocation of SI codes for Digital Video Broadcasting(DVB) systems.

[9] JavaTV™ API Technical Overview : The JavaTV API whitepaper, version 1.0