

SOA 기반의 디지털 셋탑 박스 개발

황지연*, 고흥준*, 신승중*, 류대현*
*한세대학교 IT 학부
dhryu@hansei.ac.kr

Development of SOA based Digital Settop Box

Ji-Yeon Hwang*, Hyung-Jun Kho*, Seung-Jung Shin*, Dae-Hyun Ryu*
*Dept of IT, Hansei University

요 약

디지털 셋탑 박스가 보편화 되면서 다양한 콘텐츠들이 나오고 있으며, 사용자나 방송사, 사업자의 요구에 따라 웹 서비스, DLNA, FTP Server, Media, SI, 등이 변경 요구되고 있다. 본 논문에서는 웹기반의 인터넷에서 사용되는 SOA를 적용하여 기능을 서비스 단위로 재해석하고, 여러 요구조건에 대응할 수 있는 셋탑 박스 시스템을 개발하였다. 본 논문에서 개발한 SOA 기반의 셋탑 박스는 Media, SI, Cas, Demux, Subtitle, Caption, 예약 등을기능을 서비스 단위로 재해석하여 여러 요구조건에 대응할 수 있으므로 셋탑 박스 사용자, 방송사, 사업자에 의해 빠르게 변화하는 환경에 신속히 적응할 수 있다.

1. 서론

디지털 방송(Digital TV)은 기존의 방송과는 달리 시청자와 함께 호흡하는 방송으로 단방향 정보 뿐 만 아니라 상향 채널(Return Channel)을 이용한 실시간 참여, TV전자 상거래(T-commerce), 홈뱅킹(Home Banking), 전자우편, 퀴즈 등을 가능하며, 날씨 정보나 증권 정보 등 다양한 부가 정보를 얻을 수 있는 방송이다.

대표적인 디지털 방송은 유럽의 DVB, 미국의 ATSC, 일본의 ARIB 등이 있으며, 각 국은 이러한 방송 규격에 맞게 방송 서비스를 한다. 이러한 디지털 방송을 위성이나 케이블을 통해 수신하는 장치가 디지털 셋탑 박스(Digital Settop box)이다.

디지털 셋탑 박스가 보편화 되면서 다양한 콘텐츠들이 나오고 있으며, 사용자나 방송사, 사업자의 요구에 따라 웹 서비스, DLNA(Digital Living Network Alliance), FTP Server, Media, SI, 등이 변경 요구되고 있다.

본 논문에서는 웹기반의 인터넷에서 사용되는 SOA(Service Oriented Architectures)를 적용하여 기능을 서비스 단위로 재해석하고, 여러 요구조건에 대응할 수 있는 셋탑 박스 시스템을 개발하였다. 본 논문에서 개발한 SOA 기반의 셋탑 박스는 Media, SI, Cas, Demux, Subtitle, Caption, 예약 등을기능을 서비스 단위로 재해석하여 여러 요구조건에 대응할 수 있으므로 셋탑 박스 사용자, 방송사, 사업자에 의해 빠르게 변화하는 환경에 신속히 적응할 수 있다.

2. 관련기술

2.1 DVB(Digital Video Broadcasting)

DVB(Digital Video Broadcasting)이란 국제적으로 승인된 디지털 TV를 위한 공개 표준 규격이다. ATSC를 채택한 미국, 캐나다, 대한민국, 멕시코 등과 독자적인 표준을 채택한 일본을 제외한 세계의 대부분 지역에서 채택되고 있다. 한국에서도 위성방송사업자인 Sky Life TV는 DVB-S 표준으로 방송을 내보내고 있다. 방송 방식에 따라 DVB-S (위성방송), DVB-C (케이블 텔레비전), DVB-T (지상파), DVB-H(이동 전용)로 구분 한다.[1]

<표 2-1> DVB의 주요 Table name

NIT	NETWORK NAME
BAT	BOUQUET
SDT	SERVICE NAME
EIT	TIME정보, 프로그램정보
TDT	시간정보
EMM	사용자에게 SMARTCARD 정보 (CAT)
ECM	SCRAMBLED (CAT)

2.2 MPEG-2(Moving Picture Expert Group)

MPEG-2는 MPEG(Moving Picture Expert Group)이 정한 오디오와 비디오 인코딩(부호화)에 관한 일련의 표준을 말하며, ISO 표준 13818(13818-1은 시스템, 13818-2는 비디오 부호화, 13818-3은 오디오)로 공표되었다. MPEG-2는 일반적으로 디지털 위성방송, 디지털 유선방송 등의 디지털 방송을 위한 오디오와 비디오 정보 전송

을 위해 쓰이고 있다. 또, MPEG-2의 표준을 약간 변형한 인코딩 포맷은 상업 DVD의 표준으로 돌비 디지털, DTS와 함께 사용되고 있다.

MPEG-2 13818-2 비디오 표준은 MPEG-1과 비슷하지만, 텔레비전 방송에서 사용하는 비월주사 방식의 영상을 지원한다. MPEG-2 비디오(부분 2)는 저속 비트율(1 Mbit/s) 환경에는 부적합하지만, 초당 3 메가비트 이상을 요구하는 MPEG-1보다는 향상된 압축률을 보이고 있다. MPEG-2의 MPEG-1과 구별되는 특징으로는 데이터 유실이 많은 전송 환경에도 적합한 트랜스포트 스트림이 정의되어 있다는 점을 들 수 있으며, 이는 현재 디지털 방송에 사용되고 있다. MPEG-2는 원래 MPEG-3로 개발하려던 HDTV(고선명 텔레비전) 전송의 표준 또한 포함한다. 또한 MPEG-1과도 호환성도 보장되어 표준을 따르는 MPEG-2 디코더는 MPEG-1 스트림도 재생할 수 있다. 이 부분의 표준은 ITU-T의 비디오 코딩 전문가 그룹 (Video Coding Experts Group, VCEG)과 ISO/IEC의 동화상 전문가 그룹(Moving Picture Experts Group, MPEG)이 공동으로 표준화를 진행하였으며, 따라서 ITU-T의 H.262와 MPEG의 13818-2는 동일하다.[2]

<표 2-2> MPEG-2의 주요 Table name

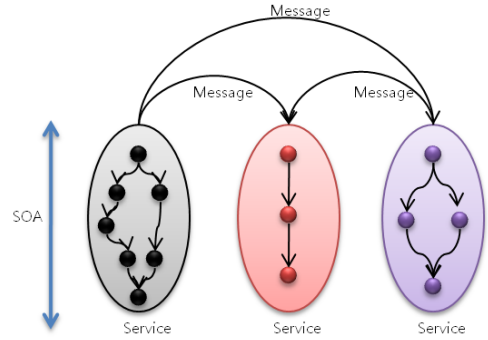
PAT	n개의 PROGRAM
CAT	SCRAMBLE
PMT	STREAM TYPE PID

2.3 SOA(Service Oriented Architectures)

서비스 지향 개념을 기반으로 소프트웨어 시스템을 구축하고자 하는 노력을 총칭하여 서비스 지향 아키텍처 (service-oriented architecture, SOA)라고 한다. 서비스 지향 아키텍처는 비즈니스 프로세스와 그것을 지원하는 IT 기반 구조를 안전하고 표준화된 컴포넌트-서비스로 통합하기 위한 프레임워크이며, 이들 서비스는 변화하는 비즈니스 우선순위를 해결하기 위해 재사용하고 결합된다.[3]

SOA에 대한 가장 많은 정의는 아키텍처와 구현 기술 관점에서 바라보는 것이다. 아키텍처 관점에서는 개방형 표준(open standard)을 따라 자치적인 서비스(autonomous service)가 그림2-1과 같이 메시지 기반(message-based)으로 서로 커뮤니케이션하도록 소프트웨어 시스템을 설계하는 방식을 정의하는 아키텍처적인 원리와 패턴을 포함하는 일종의 아키텍처 스타일(architectural style)로 SOA를 정의한다.

서비스란 생산된 재화를 운반, 배급하거나 생산이나 소비에 필요한 노무를 제공하는 것을 의미하며, SOA에서는 하나의 컴포넌트가 다른 컴포넌트와 인터페이스 계약을 통해 제공되는 행동으로 정의한다. SOA의 요구사항은 Layered, Rule, MVC, Service Hierarchy, Component, Class, Function, Data, Reuse을 정의하고 있다. [4]



(그림 2-1) Message방식의 SOA의 구조

2.4 UDS(Unix domain socket)

UDS는 socket API를 수정 없이 이용가능하며, port 기반의 Internet Domain Socket에 비해서 로컬 시스템의 파일시스템을 이용해서 내부프로세스간의 통신을 위해 사용한다는 점이 다르다고 할 수 있다. 파일을 통해서 통신을 하며, 커널내부에서 메시지를 관리한다는 점에서 FIFO와 매우 유사한 면을 보여주지만, FIFO와는 달리 양방향 통신이 가능하다는 특징을 가지고 있다. 그러므로 다중의 클라이언트를 받아들이는 서버/클라이언트 모델을 만들기가 매우 쉽다. 또한 Inet 소켓을 통한 외부통신에 비해서 2배 이상의 효율을 보여준다는 장점을 가지고 있다.

2.5 Shared Memory

모든 프로세스는 자신의 업무를 수행하기 위해서 필요한 자료를 저장하기 위한 메모리 공간을 가지게 된다. 이러한 메모리공간에는 CPU에 의해 수행되는 명령어들, 프로그램 시작시 정의되고 초기화된 데이터, 프로그램 시작시 정의되었지만 초기화 되지 않은 데이터, 함수호출에 필요한 정보, 동적할당이 이루어지는 데이터등 이 들어가게 된다.

프로세스는 시작시 혹은 실행중에 이러한 데이터를 저장하고 사용하기 위한 메모리 공간을 커널에 요구하여서 할당받아 사용하게 되는데, 이러한 메모리공간은 기본적으로 메모리를 요청한 프로세스만이 접근가능하도록 되어있다. 하지만 가끔은 여러개의 프로세스가 특정 메모리 공간을 동시에 접근해야할 필요성을 가질때가 있을 것이다.

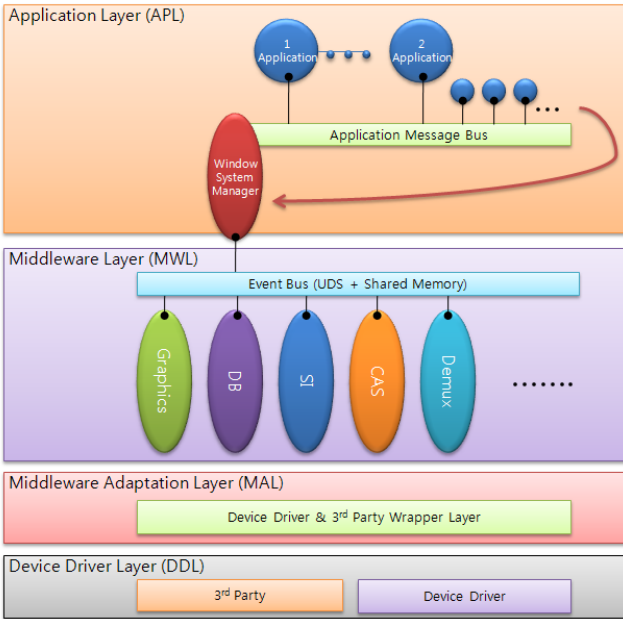
공유메모리는 여러 IPC 중에서 가장 빠른 수행속도를 보여준다.

그 이유는 하나의 메모리를 공유해서 접근하게 되므로, 데이터 복사와 같은 불필요한 오버헤드가 발생하지 않기 때문으로, 빠른 데이터의 이용이 가능하다. 그러나 하나의 프로세스가 메모리에 접근중에 있을때, 또다른 프로세스가 메모리에 접근하는 일이 발생하면 자칫 데이터가 훼손될 수 있을것이므로, 한번에 하나의 프로세스가 메모리에 접근하고 있다는걸 보증해줄 수 있어야 한다.

3. 설계 및 구현

3.1 시스템 구조

본 연구에서는 기존 디지털 셋탑 박스의 기능에서 서비스 단위로 재해석 하고, 서비스간의 독립성을 위해 Linux 기반의 UDS를 사용하였으며, Settop의 대용량 데이터를 처리를 위한 Shared memory를 사용하였다. 수신 모듈로는 MPEG-2 Demux 칩이 있는 임베디드 컴퓨터를 사용하였다. Settop System을 보다 효율적으로 적용하기 위해 4개의 Layer로 나누어 구현하였다.



(그림 3-1) SOA를 기반으로 한 Settop 구조

3.1.1 APL(Application Layer)

APL은 화면에 보여지는 Application이 있는 부분이다. 가장 많은 변화가 가능하고, WSM(Window System Manager)가 있다. Application은 WSM을 통해 Event Event Bus에 Event를 요청할 수 있으며, MWL에서 APL로 올라가는 데이터는 UDS Event로 알려주고, 해당 데이터는 Shared Memroy에 들어 있다.

3.1.2 MWL(Middleware Layer)

MWL은 Event 통신을 위해 UDS를 사용하였으며, 각 Service의 데이터를 APL로 올려보내기 위해 Shared Memory를 사용할 수 있도록 구현하였다.

Settop에서는 Demux 칩이 존재하기 때문에 Graphics 없이, 재생이 가능해진다. 그리고 DB가 없이 화면을 보는 것도 가능하다. 이 정보들을 Service 관점에서 보고 SOA를 기반으로 구성하였으며, DVB에서 ARIB가 변경되는 시점에는 SI만 변경하면 사용 가능한 시스템이다. 이를 더 활용하여 DVB와 ARIB를 동시에 연결하고, 사용하는 방송 시스템마다 필요한 것만 연결하는 방식을 통해 시스템이 느려지지 않고도 구성이 가능하다.

DB를 사용하지 않고 EEPROM이나 Flash에 직접 Read/Write를 한다면, DB 모듈 대신 EEPROM이나 Flash 모듈로 변경하면 되도록 구현 하였다.

3.1.3 MAL(Middleware Adaptation Layer)

MAL의 경우 MWL의 독립성을 보장하기 위해 존재한다. 3rd Party나 Device Driver의 버전이나 API 변경으로 인하여 MWL이 영향을 받지 않게 하기 위해 구현하였다.

3.1.4 DDL(Device Driver Layer)

Device Driver와 외부 Solution을 이용하는 3rd Party로 구분된다. Linux Kernel Version이나 Chip 변경으로 인하여 H/W Setting이 변경되는 경우를 보장하기 위해 구현 하였다.

3.2 시스템 Message

3.2.1 APL Bus Message

APL Message는 WSM에서 Event Bus에서 온 데이터를 알려준다.

```

APL_Result_t AP_Wzd_Main_Proc
{
    switch (nMessage)
    {
        case WSM_CREATE:
        case WSM_KEYUP:
        case WSM_KEYDOWN:
        case WSM_CLICKED:
    }
}
    
```

(그림 3-2) APL Bus Message

3.2.2 APL의 Window Manager Loop

Window Message는 10m Sleep time을 두고 Message가 들어오지 않을 경우 화면을 갱신하도록 하였다.

```

ulMessage = MWL_EventBus_ReceiveTimeout
(&msg, sizeof(MSG), WAIT_10MS);
switch(ulMessage)
{
    case MWL_EVENT_PAINT:
    case MWL_EVENT_NO_EVENT:
        /* 화면을 갱신한다. */
    case MWL_EVENT_ALARM:
        /* 알람을 울린다.-MWL의 Timer */
    case MWL_EVENT_TIMER:
        /* Timer를 발생한다-APL의 Timer */
    }
}
    
```

(그림 3-2) APL의 Window Manager Loop

4. 결론

디지털 셋탑 시스템에 SOA를 기반으로 구현함으로써, 유럽이나 한국, 일본, 미국에서 사용가능한 유연하게 대처할 수 있도록 하였다.

하지만 이 시스템에서 해결해야 할 몇 가지 문제점으로 UDS Event 방식을 사용하기 때문에 Linux 기반으로 생산된 셋탑박스만 사용가능하다는 점, CAS와 SI간의 통신을 할 경우 지연 처리되어 CAS Message가 늦게 올라오거나 SI가 빠르게 반응하지 못하는 점, DDL Layer와 MAL Layer의 역할이 좀더 세분화 되지 못해 중복 구현 또는 잘못 구현될 수 있다는 점, 빠른 EIT처리에서 다른 셋탑보다 느려질 수 있다는 점 등이 있다.

본 연구에서는 디지털 셋탑의 기능을 유연하게 변경가능 하도록 SOA방식으로 접근하였고, 이를 위해 기능을 서비스 개념으로 변경하여 설계 및 개발 하였다.

그러나 UDS Event 방식으로 변경되면서 기존에 Kernal Message를 사용하는 것 보다 약 50ns정도의 시간이 더 소요되는 문제점이 존재한다. 이 분에 대해서는 향후 연구되어야 할 것이다.

참고문헌

- [1] wikipedia : <http://ko.wikipedia.org/wiki/DVB>
- [2] wikipedia : <http://ko.wikipedia.org/wiki/MPEG-2>
- [3] Service-Oriented Architecture Compass(2006, Norbert Bieberstein 외, IBM press)
- [4] 이 강표 외, SOA 기반의 U-City 서비스 통합 아키텍처, 정보과학회논문지 : 컴퓨팅의 실제 및 레터 제16권 제3호, 2010.3 : 257~375(119pages)
- [5] 한국표준과학연구원 <http://www.kriss.re.kr/>
- [6] 소프트웨어 아키텍처: 이론과 실제(2007, 랜 베스 외, 에이콘)
- [7] 클라우드컴퓨팅 애플리케이션 아키텍처: IT 서비스의 미래 비전 (2010, 조지 리스, 지앤선)